# ITIL KB

# Documentation

Published: April 21, 2014

# Table of Contents

# PROJECT STATEMENT

## INTRODUCTION

The ITIL KnowledgeBase provides all the functionality to manage a complex IT organization, including Service Request Management, Incident Management, Problem Management, Change Management, Configuration Management, Purchase Management, Project Management, and more.

Our goal has been to provide ready-to-go, out-of-the-box ITIL-compliant structures and process flows for managing IT services, while enabling complete extensibility to meet the needs of any particular organization.

ITIL is a set of best practices intended to improve IT service while reducing failures and costs.  The actual implementation of any ITIL process is open to a wide range of interpretations.  While the basic structure of Service, Incident, Problem,  Change, and Configuration Management is likely to be part of any ITIL implementation, which services fall under which category, what the relationship of the Service Catalog is to these various processes, how these requests are structured and managed, and many other details will vary widely based on the needs and preferences of a particular organization, as well as the functionality of the particular software program used to implement ITIL.

Companies may implement ITIL along a spectrum moving from simple to complex -- from a fairly streamlined, direct approach, to a more "controlled" approach, with more process steps and approvals required for relatively simple tasks.  Based on our years of experience implementing IT service solutions that people actually use, we have designed our out-of-the-box implementation to keep things as simple as possible within the ITIL framework.

For instance, we have chosen to put requests for standard changes that require no special approvals into the Service Request table rather than the Change request table, keeping the workflows for each request type more distinct and simpler.   Services such as New Employee Setup and Password Resets are therefore handled within Service Requests.

We have also designed the system so that if a person has a problem with his printer, it does not **necessarily** require the creation of an incident, a problem, and a change request, just to get a new ink cartridge installed, something that at least one competitor has defined as the "correct" ITIL process.  However, if you want to follow this more extended process, we have made it as easy as possible – from any record a button can be clicked to create the related records and to map field values from your current record. There are no duplicate text entries when creating problems, change requests and incidents.

Since there are so many different ways of implementing ITIL, this document is a detailed guide to how it has been implemented in the out-of-the-box Agiloft ITIL KB.  We have pre-built the complex relationships and functions that many companies may want, while trying not to force too much complexity on those who may prefer a more nimble and efficient implementation.  This is a rather difficult balancing act, and while we have done our best to get it right for the largest number of customers, the real power of the system is in how easy it is to change it to adapt to your company's specific preferences and needs.

We will offer some guidance throughout this document to where to go to make changes to the critical relationships and behavior to suit your needs.

# GROUPS AND TEAMS

A first step in defining your processes and customizing the system is to consider the different sets of users who will be using the application and what kinds of access they will need.

Users in Agiloft belong simultaneously to both groups and teams.  A user can belong to multiple groups (receiving the superset of those groups' permissions) and to a primary team with additional teams.  A user's access to the system – the tables and tabs he sees, the records he sees, the fields he sees, the records he can create and edit, and the menu actions he can perform - depends on his group memberships.  While you can create as many groups as you need, it is preferable from a maintenance perspective to keep the number of groups small.

A user's primary team determines what look and feel scheme he sees – so you can have customers on different teams actually seeing a differently branded interface with different logos and colors.  Staff Teams are generally used to define functional groups to whom tickets will be assigned and emails sent.

**In brief, groups determine the content of what members see. End user teams determine look and feel while staff teams define working units.**

This section describes the different sets of users and the default breakdown of users into groups and teams.

# TERMINOLOGY

A note about terminology: We use the term **end user** to mean users who access the system through the end user interface, a simplified interface that allows them to create records of any kind, view any records made available to them, edit records defined as their own, and view any FAQs made available to them.  These users cannot edit records defined as belonging to other people and they use the unlimited end user license.

We use the expressions "End User" or "Customer" interchangeably in this document to refer to company employees whose main role in the system is to make requests on their own behalf or for someone else (typically their supervisor or supervisee).

We use the term **staff** to indicate the people who are working on other people's issues – they may be solvers, technical support staff, IT staff, approvers, developers, sales reps, managers, or any other types of users who access the system through the staff interface.

"Technician" may also be used to refer to members of the IT organization or other teams that will be responsible for handling, creating, or responding to requests submitted by customers or other technicians.

Both end users and staff users may be employees of your company.  Staff users require their own named license or may share a concurrent staff license.

# GROUPS

This table lists the default groups that have been set up and describes their general permissions.  To see more information about a group's permissions, you can print out or save to a file the full details of that

group's permission by going to Setup/Access/Manage Groups, selecting the group, and mousing over the printer icon to choose the printout you want.

| Groups | Type | General description of their access permissions |
|---|---|---|
| Admin | Staff | These users can edit the system including rules and workflow, and can view, change and delete all records |
| Approvers | Staff | This group is intended to allow approval permissions to the Contract table, and tables the Contract table is dependent on, to any user it is applied to. They can view and edit Contracts they are listed as an approver on, and can view tables necessary to approve Contracts such as Steps, Approvals, Approval Templates, and Companies. |
| Base ServiceDesk | Staff | This group has the base permissions that should apply to all more privileged groups dealing with the Service desk tables.  Users in those groups should also be made a member of the Base Servicedesk group.  This group has full create/edit access to records in the Service Request, Incident, Problems, and Tasks tables, and create/edit own access to Change Requests and time entries. It has full view access to Configuration Items, Services, Companies, and Employees and can edit its own employee record, but has no other create or edit access in those tables.  It can create/edit end users (external customers).  It cannot delete records. |
| Document Creator | End User | Can create documents and edit their own – customer of document table. |
| Document Manager | Staff | People who can approve and publish documents. Cannot modify status of document records manually. |
| Document Reviewer | Staff | Can edit approvals for which they are the approver. |
| Service Manager | Staff | For staff responsible for maintaining the Service Portfolio (Service table) and the Tasks/Templates table. |
| Change Manager | staff | This group is responsible for management of Change Request records and has full privileges on the Change Request table.  Members can create, edit, and delete records in this table and will typically be Change Managers or Change Owners. |
| Change Approver | Staff | This group includes members of the Change Advisory Board (CAB) and Emergency CAB as well as the Change Management Team acting as Change approvers.  This group has edit access to all approval fields, while no other groups can approve changes. |
| Configuration Manager | staff | This group has full access to the CI records and is responsible for creating, editing, and deleting those records.  People responsible for working on and configuring CIs, managing CI resources, and so on would typically be in this group. |

| Procurement Group | Staff | This group is responsible for managing the Purchase Request and Item tables. |
|---|---|---|
| Internal customer | End user | Internal Customer in employee table, can request Service Requests, Purchase Requests, and report Incidents, as well as see their own Configuration Items and edit some of their profile information and view other employee contact information. They may also access Knowledge FAQs. |
| Contract Manager | Staff | This group has full access to the Contract table, Approvals table, Steps and Workflow tables, and Companies table. They also have some access to End Users and Employees. They are responsible for creating, editing, and approving contracts for customers or the company. |
| Contract Owner | Staff | This group has a subset of the permissions Contract Managers do. They are responsible for Contracts assigned to them, and have full permissions there, but can only view Contracts that they did not create or were not assigned to. They have all the other permissions necessary to allow them to use Contracts effectively. |
| Customer Manager | Staff | Customer Manager – unused – relevant if providing external customer support.  Customer managers can view all support cases for their own company. |
| Customer | End user | Unused unless providing external customer support.  Then this group is used for end user customers, who can submit and view their own support cases. |
| Sales | Staff | This group is responsible for recording information regarding sales efforts to specific companies as well as Purchase Orders made. They can also create and update Support Case records for the companies they represent. They have full access to: Company, Contract, Lead, Opportunity, and PO tables. Partial access to Campaign, Product, Product Quoted, Project, Quote, Support Case, Tasks, Teams, Time Entry, People: End Users. |
| Marketing | Staff | This group is responsible for coordinating and recording information about marketing campaigns and providing quotes to prospective customers. They have full access to: Campaign, Company, Lead, Opportunity, and Product tables. They have some access to Product Quoted, Quote, Tasks, Teams, Time Entry, People: End User, and People: Employee tables. |
| Guest | End user | This group may be used in hyperlinks to allow creation of new requests of any kind (such as leads, users, Incidents) without seeing the rest of the user interface |
| Anonymous | Staff | This group is used to enable unregistered users to click on an email hyperlink sent in an outbound email to be able to edit that record.  It is used in conjunction with the Anonymous user.  If all your users reside in the system, you will not need this group. |

# TEAMS

Teams serve a different purpose for customers/end users and for staff users.  For end users, unless you a) need to provide multiple branded interfaces to different sets of users, or b) expect your end users to access the system in multiple languages, it is simplest to put them all in one team with the word customer in the title, such as Customer Team or Internal Customer Team.

If you need to provide a custom look and feel and customer branding to different sets of users, then you will need a separate customer team to go with each "look and feel scheme".  Teams can also be associated with a different default language, so it would make sense to have language-based customer teams if you were planning to run the program in a multi-lingual environment.

For staff users, teams are used to identify the functional units to whom records may be assigned.  You will want a staff team for each assignment group (sometimes called a Queue).  Teams can be hierarchical, so you may set up a hierarchy such that you can send an email to a mid-level team and the members of its subteams will receive the email.  Users have one primary team, which defines the look and feel they see and their default table views.  They can be members of as many additional teams as needed, so they are cc'd on emails and included in the assignment list for items assignable to those teams.

## TEAMS TABLE IS SPECIAL

The Teams table is a system table with some special hard-coded fields, such as the Working Hours and language fields, the team name, label, description, Team Leader, and some screen refresh options.

It also has some custom fields that can be modified, and you can add your own custom fields to the teams table to manage any other information associated with your teams.  If you edit the teams table setup, you will only be able to modify the custom fields and their layout options.

Creating new teams and accessing all of their attributes is done through the Setup/Access/Manage Teams screen.  Editing just the custom fields and adding users may also be done through the Teams tab on the left toolbar.

## MANAGING TEAM MEMBERSHIP

Team membership is controlled by two fields in the user record that are linked to the Teams Table: Primary Team (single choice), and Teams (multi-choice).  We recommend adding a user's Primary Team to the multi-choice Teams field along with any additional teams.  This way a user's entire team membership can be found in a single field, which makes searching and filtering easier.

When you initially import users into the system, you can import their primary team and teams values (with multiple values separated by , with no spaces).  Once users are in the system, you may update team membership by editing specific users or by editing the team through the Teams tab, and looking up and importing users for the embedded Primary Team Members or Team Members tables shown for the Team.  Looking up and selecting a user for either of these tables changes the linked field in the user's own user record to point to this team.

## DEFAULT TEAMS

The teams shown below have been set up as the default teams for the application.  You can delete any teams you do not need and you can rename any team to match your own company's naming

conventions.  You can of course also create any additional teams you need.  This is best done by going to Setup / Access / Manage Teams and either editing a team or creating a new one there.

| Teams | Description |
|---|---|
| Project Manager Team | Project Manager Team |
| Document Reviewers Team | People who have been identified to review documents. |
| Document Management Team | Team for Document Management |
| Contract Owner Team | For staff contract owners/buyers |
| Finance Team | Finance team used in contract approvals |
| Compliance Team | Compliance Team used in contract approvals |
| Risk Team | Risk team approves contracts |
| Legal Team | Legal Team approves contracts |
| Vendor Team | Container for external Vendors - used in Contract Management |
| Knowledge Team | Knowledge Team for publishing FAQs |
| Office Mgmt Team | Office mgmt support team |
| Document Creator Team | Document Creator Team for internal end users. If using external users, then change the parent team. |
| Facilities Team | Facilities support team |
| HR Team | Human Resources Team. Acts on new employee Service Request Tasks. |
| Vendor Management Team | Vendor Management Team |
| Contract Management Team | Contract Management Team |
| Desktop Applications Team | CI Desktop and Support Team |
| Purchasing Team | Purchasing Team |
| Security Team | CI Security and Support and Change Team |
| Database Team | CI Database Team |
| Backup and Storage Team | CI Backup and Storage Team |
| System Administration | CI Sys Admin Team |

| | |
|---|---|
| Team | |
| Custom Applications Team | CI Software Team |
| Server Team | CI Server Team |
| Network Operations Team | CI Network Team |
| Configuration Management Team | CI Team |
| Change Approver Team | Change Approver Team |
| Change Management Team | Change Team |
| Professional Services Team | will be assigned to projects |
| Admin Team | Used for system notifications about rule, email and other errors |
| 2nd Level Support Team | Second Level Support Team |
| Sales Team | Sales Team |
| 1st Level Support Team | First Level Support Team |
| Marketing Team | Marketing Team |
| Customer Team | External Customer Team |
| Internal Customer Team | Internal Customer Team |

# DEFAULT CONFIGURATION OF TABLES AND PROCESSES

## REQUEST TABLES OVERVIEW

Deciding how to segment the service catalog and what table structure to use for an ITIL implementation can be quite a challenge.  In defining the default table structure, we had a few basic goals:

1) To make it easy for end users (the customers) to know where to go to request what they need and to follow up on their requests
2) To make the backend processing and workflows as simple and clear cut as possible for the technical users and approvers who must process the requests
3) To make the system easy for administrators to understand and maintain so that changes may be easily made to one kind of request and how it is processed without affecting everything else

There is a certain tension between these goals, but happily, we believe they are all accomplished in our default setup.

We have chosen to segment customer requests into a few distinct request types and tables illustrated in the diagram:

```
                    ┌─────────────────────────────────────┐
                    │       End User Point of Entry        │
                    └─────────────────────────────────────┘

    ┌─────────────────────────────────┐            ┌──────────────┐
    │    Service Catalog (Services)    │            │   Incident   │
    └─────────────────────────────────┘            └──────────────┘

  ┌──────────┐   ┌──────────┐   ┌──────────┐        ┌──────────┐
  │ Service  │   │ Purchase │   │ Change   │        │ Problem  │
  │ Request  │   │ Request  │   │ Request  │        │          │
  └──────────┘   └──────────┘   └──────────┘        └──────────┘
```
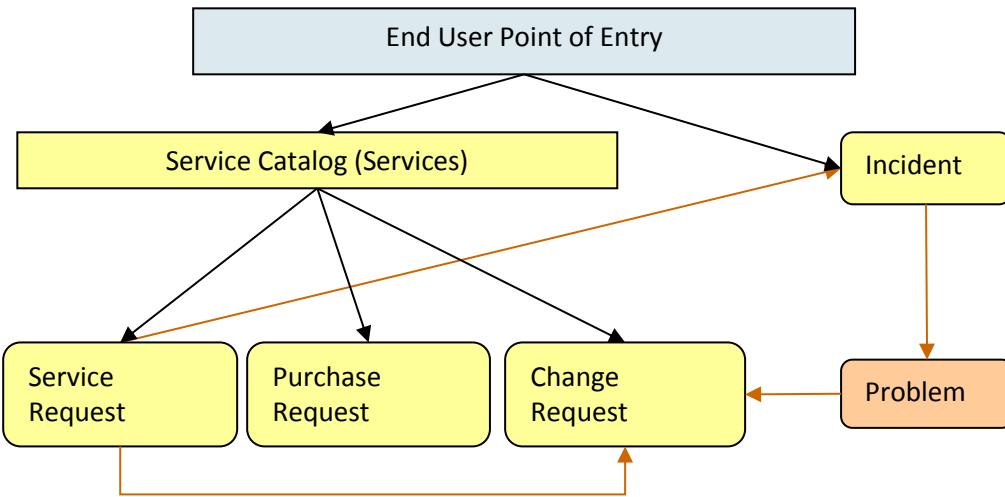
The service catalog offers services to the end user that may be created in one of three tables: Service Requests, Purchase Requests, or Change Requests.  Whether or not a particular group of users can request particular services can depend on their group permissions – for instance, our default internal end users cannot directly create a Change Request, while other technical users can.  An end user may also submit an incident to report a problem or interruption to a business or technical service.  If merited, that incident may lead to the creation of a Problem and a Change Request by technical staff.

The system may be configured so that end users simply create records directly in any of these tables and choose the appropriate service from the service catalog once in the record, or it may be set up to offer users a set of hotlinks to create requests for specific services without them needing to know what table will be used to store and process the requests until after they have created them.

The separation of requests into these tables makes maintenance and processing simpler than if they were all in a single table, since the fields, form layouts, access permissions, workflows and rules can be specific to the needs of the particular request type.

The orange arrows in the diagram above show the predefined conversion paths a request may easily take, if a technician believes it is appropriate – a service request can, with a click of a button, generate an Incident or a Change Request, or can be linked to existing incidents or changes.  Likewise, an Incident can generate or link to an existing Problem, while a Problem can generate or link to a Change Request.

A Problem may be related to multiple Incidents and a Change Request may be related to multiple Problems, Service Requests, and (through Problems) Incidents.

Naturally there are relationships within each of these tables to Users, Teams, Configuration Items, and other essential components of the system.

## SERVICE CATALOG – SERVICES TABLE

The Services table is the source for the ITIL Service Catalog.  It holds a record for each service that may be offered to end users through the service catalog.  All services offered by the organization to end users should be managed here and this table's workflow is used to manage the approval process for a new service.

Each service is managed by a different request type, such as a Service Request, Purchase Request, or Change Request.  Each service is therefore associated with one of these three tables, and this association is defined in the Services field "Show in Table".

There are two ways to request a particular service – one is to create a new request of the correct type, such as a Service Request, and then to pick the service category and then the service from the drop-down list of services.  The second method is to use an internal hotlink such as "Request a Password Reset" which will take the user to a new record in the appropriate table with the Service automatically set to password reset.

We have created some sample services and predefined which type of request they are associated with.  Naturally, your organization will want to create its own services, delete some of our sample services, or change the associations we have made for those services.

For instance, the onboarding of new employees is currently managed through the Service Request table under the Service Category New Employee Setup.  Some organizations may prefer to manage new employee setup through the Change Request table instead.  Making this kind of change is a matter of modifying the service record and changing the field called "Show in Table" to point to the desired table.

Since the Service Catalog is the backbone for many ITIL processes, the Services table has several special fields that are pulled into the corresponding request to enable automation, escalation, and special field visibilities and dependencies based on the type of service.

Note that in an effort to allow simplicity, we have not added a table to hold different SLAs, but have simply added some simple SLA fields to the services table.  For organizations with a variety of complex SLAs based on multi-field criteria, it will make sense to create a separate table to manage all SLAs and then to pull in the appropriate SLA record for a particular service.

Following is a picture of the default form for Services.  Below that is a description of all of the default fields for the Services table.  Note that field explanations are also available onscreen while working with the Services input form by clicking on the underlined field labels:

Choose multiple categories for this service.

Choose primary category for this service.

## FIELDS FOR SERVICES TABLE

Fields are described in the Services Fields section of the Appendix.

## SERVICE CATALOG MANAGEMENT

Once the Service Catalog is initially set up by creating the appropriate services in the Service table, new service items will be created by staff members only when a new service is planned or initiated.  Existing

services may naturally be modified.  Initial permissions are set so only users in the admin group can create/modify/delete services.

Service records may be created in a state of Planned or Active.  Once all active services have been set up during the initial system configuration, it may be desirable to edit the workflow to deselect the Creatable box for the Active State, so that future services must start in a status of Planned and go through an approval process.  Temporarily inactive services should be placed in the Inactive state.  Services no longer in use can be placed in the Retired state for record-keeping.

## Ownership

Services are technically owned by the Responsible team for that service.    However, only the admin group has edit permissions by default.

## Workflow
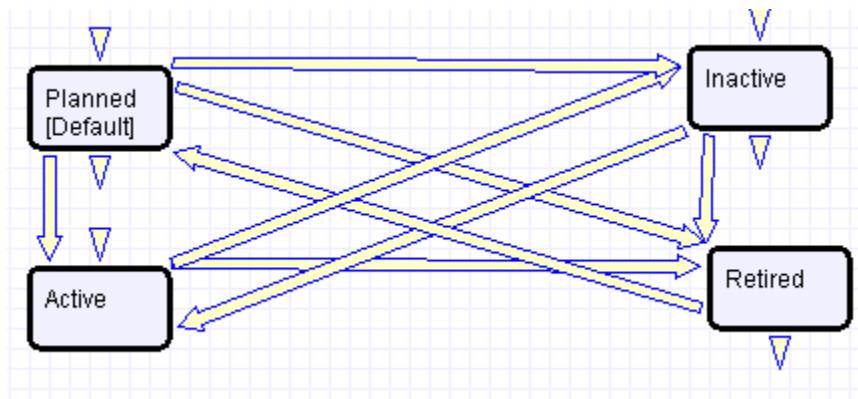


## SAVED SEARCHES

Default searches provided are detailed below.

| Saved Search Name | Search Description |
|---|---|
| Service Pipeline | Status is Planned |
| Service Catalog | Status is Active |
| Inactive Services | Status is Inactive |
| Retired Services | Status is Retired |

## SERVICE REQUESTS TABLE

## OVERVIEW

The Service Requests table holds all customer service requests.  A service request (SR) is created when an internal customer needs a new service, a standard (pre-approved) change to an existing service, a password reset, or some other kind of assistance or information.

When the user needs to report an interruption in service or a problem with an existing service, an Incident should be created instead.  We have chosen to separate service requests from incidents because these two kinds of request have rather different workflows and resolution criteria.  An incident may often require the generation of a Problem and a Change Request, while Service Requests will generally be self-sufficient and relatively quick to resolve.

The easy way to distinguish Service Requests from Incidents is to recognize that Service Requests are the kinds of things you will choose from a service catalog, such as application assistance, password resets, new employee setup, documentation or copying services, and so on.

If a user creates a Service Request when he should have created an Incident, the technician may click a button in the SR form to easily convert that Service Request into an Incident and have all the fields mapped appropriately.

The Service Request table uses the same choice list of service categories as that used in the services table.  A user creating a service request chooses the category of service from a drop-down, and then sees all the active services that belong to that category in a second drop-down list.  The default service categories for service requests can be seen in the fields_and_choices.xls worksheet (or by creating a SR record and opening the drop-down list).

## USE CASE

Service Requests (SR's) may be created by telephone support staff or technicians on behalf of customers or by customers directly through the web interface or by inbound email, should an inbound email account be set up.

## End User Record Submission

When a customer submits a Service Request, the contact information fields automatically populate based on the details in his/her User record.  In the default setup, a request cannot be submitted on

behalf of a user who has no user record in the system.  This may easily be changed by modifying the linked fields from the User table to enable "non-source" values to be used.

The user is required to select a service after selecting a service category.  When the service is selected, the user will see a description of the service, any special instructions for that service, and any additional dependent fields defined to be visible for that service.

He may then provide further information and click Finish to save the form, after which he will receive an email acknowledging receipt of the request.

## Automatic Assignment of New Requests

When a new request is saved, it is automatically assigned to a team based on the values in three fields pulled from other tables (Rule: All new Service request actions, Action: Assign Service Request).  The related service has a Responsible Team associated with it and a field called "Give Service Team Priority over CI Team" with a Yes/No value.  If a Configuration Item was identified for the request, then there is also a CI Responsible Team field that will be populated in the SR.

The current assignment logic is: if the Service Request involves a specific configuration item (CI), then the request is assigned to the responsible team for that CI, unless the "Give Service Team Priority over CI Team" is set to Yes.  In this case, or If no CI is defined, then it is assigned to the Responsible team for the Service.  If there is no Responsible Team for the Service, then it is assigned to the 1st Level Support Team.

Naturally, this logic may easily be changed to suit your organization.  You can simply set a default value such as the 1st Level Support Team or you may choose some other logic to use based on different fields.  The current logic is implemented in the SR rule named "All New Service Request Actions" and may be modified there.

## Technician Record Submission

If a technician creates the SR, he may choose an assigned team and assigned person from the drop-down fields, and his changes will not be overwritten – the rule only assigns if the Assigned Team field is empty when the record is saved.

New SRs are created in a Status of Open by default.  During creation, a technician may change the Status to In Progress (if he wishes to indicate he is already starting work on it) or Closed (if he was able to resolve the issue over the phone and wants to capture the request for reporting purposes).

## Automatic Emails Sent upon Submission

If a Support Staff technician creates a record in a status of Closed an email is sent to the customer telling her how to reopen the Service Request.  This email is sent by a workflow action and is displayed as a checkbox that can be turned off by technician users.  (This option is set in the workflow options, and can be modified).

If the status is not Closed when saved,  the rule named "All New Service Request Actions" will send the customer an acknowledgement email and will also send an email to either the Assigned Person, if there is one, or to the Assigned Team.

## Special Handling for the New Employee Setup Service

One of the services that may be requested is for New Employee Setup.  Typically, onboarding a new employee may require the completion of several standard tasks, such as providing office supplies, setting up a computer, phone, desk, etc.  These tasks may be done by completely separate teams in parallel.

We have predefined a method of generating these tasks when creating a request for this service.  It has the benefit of flexibility - there is a multi-select field called New Tasks that appears when this service is selected.  The submitter may check just the tasks that need to be done for this new employee.

When she saves the request, individual task records will be auto-created and shown in an embedded table called Tasks within the SR.

The items shown in the New Tasks fields are actually linked records held in the Tasks/Templates subtable.  So if the tasks your company typically needs to do for a new employee are different from the ones we have predefined, you can edit the existing task templates or create new ones.

To work with records in that table, expand the Tasks table in the left pane and click on the Templates subtable. To see the task templates designated for new employee setup, mouse over the saved search link and click the search called F: New Employee Setup.   This search finds records with a value of New Employee Setup in the "Related Service" field and Active in the Template Status field of the template record.  So to eliminate tasks you don't need, you can simply delete them, or edit the Task/template record and change the Template Status field to Inactive.  To add a new template, simply create records in this Task/Template sub table and set the Related Service field to New Employee Setup.  The Task summary field is what appears in the SR multi-select field.

The creation of the related tasks based on the user's selection is done by an action called "I: Trigger creation of tasks from template" that is run as part of the "All new Service Request Actions" rule.  The rule actions running this are a bit complex and we recommend seeking our consulting assistance if you would like to modify or extend this functionality.  But if you simply want to turn it off, you can edit the rule and remove this action from the rule.

When tasks are created, two fields track the number of tasks and the number of completed tasks (those with a status of Done).  When these two fields match, meaning that the final task has been completed, a rule notifies the assigned person or assigned team (if no assigned person) that the final task is done (Rule: Tasks Just Completed: Action: Notify Assigned team or person of completed tasks).

## Processing of Records

When a technician works on a Service Request, if he needs more information from the customer in order to take further action, he can set the status to Pending Customer.  A workflow action will automatically send an email to the customer requesting further information and including the content of the Additional Notes field, an append-only field that is used to communicate with the customer.  The email includes a hyperlink for the customer to click to edit the Service Request directly.

If the technician needs to reassign the Service Request to a different team or person, he or she simply changes the Assigned Team and/or Assigned Person field and the system will email the new assignee notifying them of the reassignment (Rule: Assigned team or person changed, Action: I: Notify team or person of new assignment)

The Staff Only Notes field is an append-only field that holds technician working notes that are not visible to the customer.

When the technician has completed work on the Service Request, he/she sets the Status field to Closed and puts the solution notes into the Solution field. This triggers a workflow email to the customer that includes the content of the Solution field and tells the customer that the request has been completed. By default no escalation rules are set up for the Service Request table.

## Reporting Time Spent

Technician users may easily report the time they spend on handling service requests. There are two fields and an action button: Time Spent, Time Description, and Add Time on the Working Notes tab of the layout. Entering values there will automatically create a new time entry record when the Add Time button is clicked. The time entry will show the work done by the technician and on the current date.



All the time entries for a request can be seen on the Time tab as well. If a technician needs to report time that was spent on a different day or by a different user, he may click the New button on the Time spent table to submit a time entry directly and change the date or "Done by" field. All time entered is totaled in the All Time Spent field, which can be used in reporting or billing.

The Add Time action button converts the time fields into a Time Entry record and then blanks out the current values so they will be empty.

Note that this same time entry methodology is used in the Incident, Problem, Change Request, and Task tables.  So time spent on each type of request will be held in a single table (Time Entries) from which reports may easily be run for all kinds of time.

## Customer Updates

If the customer updates the Service Request at any point, an email notifies the assigned person or assigned team (if no assigned person) of the update.

When the customer replies to the email or edits a Service Request in a status of Pending Customer, the status changes to Updated by Customer (Rule: all customer update actions, Action: User Update Actions) and an email notifies the assigned person or the assigned team (if assigned person is empty) that the customer has replied.  The customer is able to update the Additional Notes field directly and email replies are mapped to that same field.

The closing email to customers includes a hotlink back to the record if they wish to reopen it and instructs them to explain why they are not satisfied with the solution.  Clicking the hotlink will automatically change the "I Would Like To Reopen My Service Request" field to Yes, which in turn sets the Status of the Service Request to Reopened and notifies the assigned person (Rule: All customer update actions, Action: User Update Actions).
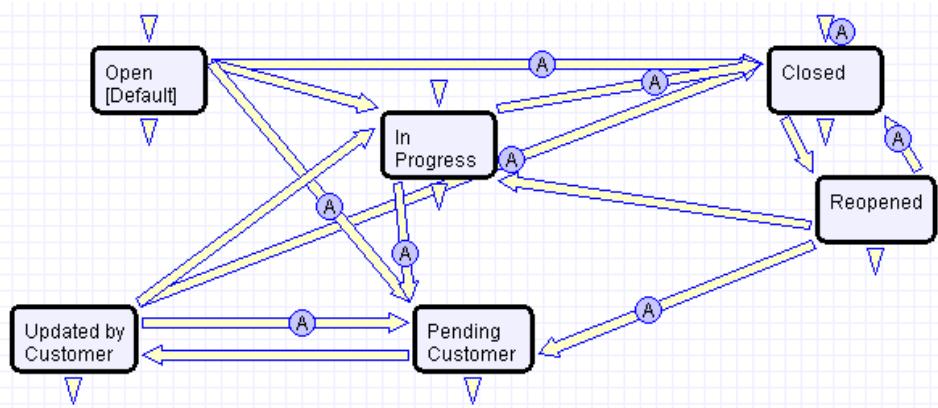
## WORKFLOW



*Service Request table default Workflow*

## FIELDS

Fields are shown in detail in the Service Request Fields table in the Appendix.

## OWNERSHIP

Records in this table are "owned" by the individual customer.  This means each record is associated with a particular employee login and no other "end user" employee will be able to edit that record.  All technician users are able to edit any service request by default.

## REPORTING AND STATISTICS

17

Several fields are included that are used for statistical reporting.  The following fields may be of interest:

Number of Assignees – is auto-incremented each time the assigned person changes (Rule: Assigned team or person changed, Action: Notify team or person of new assignment)

Number of Teams Assigned – is incremented each time a different team is put in the Assigned Team field (Rule: Assigned team or person changed, Action: Notify team or person of new assignment)

Number of Reopens – is incremented each time a closed request is reopened by a customer

Total Hours to Close – elapsed time between Date Created and Date Closed (default value)

Working Hours to Close – elapsed time between Date Created and Date Closed minus the non-working hours of the team in the Assigned Team field and the time during which the Status was Pending Customer (default value)

Solved Within SLA – yes/no field with default value of Yes, is set to No when the total hours to close is greater than the relevant SLA Hours to complete field (pulled in with the service and based on priority) (Rule: Status Change Actions, Action: All status change actions)

 There are default reports measuring Total Time Spent by Service Category, Average time to close by service category and by person who closed, as well as averages for number of people assigned, number of teams assigned, number of reopens, and so on.  With the field structures already there, it is easy to add reports to slice and dice the information the way you need it.

# TASKS TABLE

## OVERVIEW

The Tasks table holds individual tasks. While it may be used for tasks of many different kinds, it is currently set up to allow tasks to be related to Service Requests, Projects, or to be independent of other tables.  It is possible to modify the setup to relate tasks to records in any table and to show embedded tasks in any other table.

An embedded Tasks table (labeled Tasks) is shown in service requests only for New Employee Setup.  It may be opened up to visibility for any other service types that may require several assigned tasks to be done in parallel by modifying the field in the Service Request table and its visibility criteria.

If you would like Tasks to be shown in other tables or related to records in other tables, the "Related to" field in the Task table is the key field for this purpose.  The default value is currently set to Service Request. If you need to use tasks within any other table, you can modify this field and add that table to the choice list.  Then create a linked set of fields from that other table, and make these fields visibility dependent on the value in the Related to field just as we have done for the linked Service Request fields.  Then create a related table in the other table pointing to the Tasks table, and you will be able to generate tasks from there.  This advanced functionality is best done after attending a training class or with our professional consulting team's assistance.

## USE CASE

### When a Task is Created

Tasks will be created automatically on creation of a Project or Service Request that contains them. They will be auto-assigned to the appropriate team or person based on the template task record.

When a task is created, a rule called Create: All Creation Actions runs, and sets the due date. If the task is related to a service request and the Date Due is empty, then if the Template Number of Hours to Due Date field has a value (this would be auto-copied from the task template) and there is no Date Due set, and Status is not Done, and Source Template ID is not null, then the Due Date is set to the creation date plus the time given for the task in the Task Template record. Otherwise, the due date is set to 2 days from now.

If the user wishes to add time to the Task, they may fill out the Time Spent and Time Description fields, then click the Add Time button, which will convert those fields to a Time Entry linked to that Task.

In addition, notification emails are sent by the same rule – if there is an Assigned Person, an email is sent to that person, otherwise an email is sent to the Assigned Team (if there is one).  Emails are sent by the action: I: Email Assignees about new Task.


## Processing a Task

The person assigned to a task can add working notes to it, refer to the linked Service Request from it, and ultimately complete the task.  There are only 4 statuses by default: Incomplete, Done, Cancelled, and Waiting for Others.  When the person has completed the task, he may change the Status to Done and click the button "Set to Now" to populate the current date/time into the Date Done field.  He may also enter any time spent on the task in the Time Spent and Time Description fields, then click "Add Time" to convert them to Time Entries.  When a task record is saved the "Edit: Actions when closing or cancelling a task" rule is run. If the status changed to Done and nobody's name is in the Assigned Person field, the system will put the name of the last updater in that field.

## AUTOMATION AND WORKFLOW

There is a simple workflow for tasks that currently executes no actions:
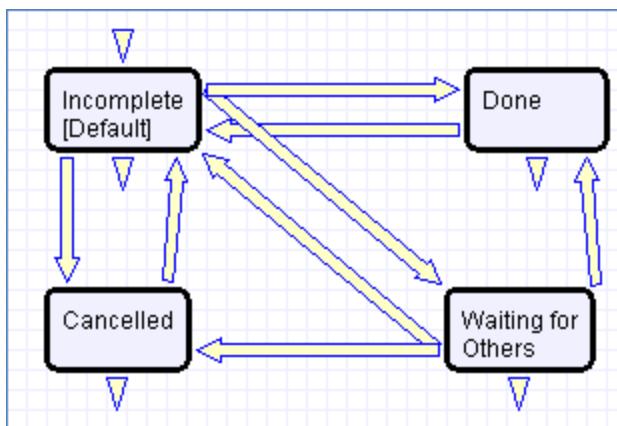


There is also an escalation rule that is set up but not running called "TB: Notify of upcoming task".  It will notify the assigned person or team when the due date is one day away, once it is turned on.  Currently it is disabled. There is a radio button at the bottom of the General tab in the table settings if this rule is desired. The schedule may need to be changed from every 10 years to something more useful.

Note that when a task related to a SR is marked to Done, the count of completed tasks in the linked SR will be automatically updated to reflect this, and when all tasks for that SR are completed, the assigned person will be notified.

## FIELDS

Fields are described in the Task Fields section of the Appendix.

## TASK TEMPLATES TABLE

## OVERVIEW

This table holds records that serve as templates for automatically generated tasks. Each Template record specifies a Task related to one other table or related to no tables.

## USE CASE

Task Templates are related to either the Project or the Service Request tables using the "Related to" field at the top of the form. Further subtypes can be selected that specify the exact type of Project or Service Request to relate the Task Template to. Task Templates related to tables in this way can be set as "Required" or "Optional" for that specific type, which affects the automation generating tasks for those tables in the expected way. Task Templates can also be "Independent", unrelated to any table.

Other tables can be added to the set that Task Templates can be related to through some modification of the automation and fields. The suggested course of action for this kind of change is consulting with a trained implementer, or using someone officially trained in the software.



Whenever a Task Template is used to generate a Task, a conversion is done from the Task Template table to the Task table, mapping important information in the Task Template to the Task record that is generated. This Task record is linked to the Service Request, Project, or other table that spawned it.

# INCIDENTS TABLE

## OVERVIEW

Incidents are used to report service problems and outages.  They are interruptions to customer service. The objective is to return to the normal service as defined in an SLA as quickly as possible with minimal business impact.  An example Incident for an end user might be: "I cannot access my email".

What is the distinction between Service Request and Incident?   Incidents are interruptions or degradations of an existing service that need to be rectified.  They are problems that need to be solved, rather than a request for new service or a change to a functioning service, such as a new employee setup or new software application request.

Incidents may result from underlying problems caused by misconfiguration or hardware failures, and an underlying problem may result in the submission of several incidents by end users.  When this is the case, the underlying problem may be reported and managed in a separate Problem record.

When an incident is caused by an underlying problem, the technician working on the incident can link the incident to an existing Problem record or convert it to create a new Problem record. They may also click the Save and Create Change Request if they wish. The Change Request table will keep track of Incidents that are converted onto it.

Problems may require significant changes in order to be permanently resolved.  In this case, the technician working on the Problem will link it to a new or existing Change Request.  While a Problem may stay open until the Change is implemented, often a workaround is found to resolve the interruption in service for the customer.

The system is set up to make it easy to generate and relate Incidents, Problems, and Changes, and to quickly and automatically propagate workarounds from problems into all related Incidents.

## USE CASE

Incidents may be created by internal customers through the web portal or via email, or by a technician taking a telephone call from a customer. They may also be created by network monitoring systems configured to send problem reports to the system through one of the standard APIs. A field in the Incident form specifies the reporting source (email, web, phone, etc.). A technician may also convert a Service Request to a new Incident via an action button in the SR form if a customer submitted a Service Request when he should really have submitted an Incident. See Details of Incident Rules in the Appendix for more details.

### End User Record Submission

When a customer submits an Incident, the contact fields automatically populate based on the details in his/her User record. In the default setup, an Incident cannot be submitted on behalf of a user who has no user record in the system.  This may easily be changed by modifying the linked fields from the User table to enable "non-source" values to be used.

The user selects a category for the Incident from the drop down list "I'm having a problem with...", such as 'Email' or 'Access Problem', then describes his/her problem in more detail via the Summary and Description fields.

The user is also given two fields to describe the priority of the Incident: Impact and Urgency. Impact specifies how widespread the Incident is: A user whose desktop computer is malfunctioning would choose 'Affects Single User', whereas a critical server failure might be classified as 'Affects Company'. Urgency is a subjective measure of the criticality and time sensitivity of the Incident, and ranges from Low to Critical. Both Impact and Urgency are used to automatically determine the official Priority of the Incident, a field which is hidden from the end user and only editable by staff.   See Incident Rules for the current logic used to set the Priority.  (Priority is only set by a rule if the staff user has not already manually set it).

After providing the summary, description, category, impact, and urgency, the user clicks Finish and receives an automatic email notification when the system creates the record.

## Technician Record Submission

A technician creating an Incident sees additional information relating to the status of the Incident, including assignee and status fields. Unlike end-users, staff can see an additional section on the details tab for Configuration Item Information that shows fields relating to a specific Configuration Item. By default Incidents are created with the "Problem CI Identified?" field set to No, but if a staff person determines the CI that is the root cause of the Incident, he or she may set this field to Yes and select a Configuration Item via a linked field set.

By default, Incidents without a CI identified are assigned to the 1st Level Support Team. Technicians can also set the assigned team and assigned person manually via drop-down lists in the Work Status tab. If the Problem CI has been identified, the staffer can set the field "Assign to CI Responsible Team" to Yes and when the Incident is saved, a rule will detect the change and assign the team responsible for the CI selected to the Incident. A technician may change the Assigned Team field later, as the rule will only set the Assigned Team at the time the CI field changes to Yes.

New Incidents are created in a status of Open by default. During creation, a technician may change the status to Assigned (if he wishes to indicate work has already started on it) or Closed (if he was able to resolve the issue over the phone and wants to capture the request for reporting purposes).

## Automatic Emails Sent upon Submission

If a Support Staff technician creates a record in a status of Closed an email is sent to the customer telling him/her how to reopen the Incident.  This email is sent by a workflow action and is displayed as a checkbox that can be turned off by technician users.  (This option is set in the workflow options).

If the status is not Closed when saved,  the rule named "Incident - All Creation Actions" will send the customer an acknowledgement email and will also send an email to either the Assigned Person, if there is one, or to the Assigned Team if not.

## Processing of Records

When a technician works on an Incident, if she needs more information from the customer in order to take further action, she can set the status to Pending Customer. A workflow action will automatically send an email to the submitter requesting further information and including the content of the Additional Information field, an append-only field that is used to communicate with the submitter. The email includes a hyperlink for the customer to click to edit the Incident directly.

If the technician needs to reassign the Incident to a different team or person, he or she simply changes the Assigned Team and/or Assigned Person field and the system will email the new assignee notifying them of the reassignment (Rule: Incident Edit Actions, Action: I: Notification Actions).

The Staff Only Notes field is an append-only field that holds technician working notes that are not visible to the customer.

When the technician has completed work on the Incident, he/she sets the Status field to Closed and puts the solution notes into the Resolution field. This triggers a workflow email to the customer that includes the content of the Resolution fields and tells the customer that the request has been completed.

## Relating Incidents to Problems

If an incident has an underlying problem that needs to be addressed, the technician may quickly convert the Incident into a related Problem record. Or she may use the Linked Problem lookup to search for existing Problems that may be the cause of this incident, and if one is found, link the Incident to that Problem.

When an Incident is linked to a new or existing Problem, the Incident may be automatically updated from the Problem record when a workaround or solution to the Problem is identified.

Within the Problem record, a button may be pressed to copy the workaround for the problem into all related Incidents, set the Status of the Incident to Workaround Provided, and email the customer and assigned rep. Another button may be pressed to populate the Problem's Solution into the related Incidents, set their Status to Closed, and email the customers.

## Reporting Time Spent

Technician users may easily report the time they spend on handling Incidents. There are two fields and an action button for this purpose (Time Spent, Time Description, and Add Time) on the Working Notes tab of the layout. Entering values there will automatically create a new time entry record when the Add Time button is clicked. The time entry will show the work done by the technician and the current date.

All the time entries for a request can be seen on the Time tab as well.  If a technician needs to report time that was spent on a different day or by a different user, he may click the New button on the Time spent table to submit a time entry directly and change the date or "Done by" field.  All time entered is totaled in the All Time Spent field, which can be used in reporting or billing.

Note that this same time entry methodology is used in the Service Request, Problem, Change Request, and Task tables.

## Customer Updates

If the customer updates the Incident at any point, an email notifies the assigned person or assigned team (if no assigned person) of the update.

When the customer replies to the email or edits an Incident in a status of Pending Customer, the status changes to Updated by Customer (Rule: Incident Customer Update Actions, Action: Submitter Update Actions) and an email notifies the assigned person or the assigned team (if assigned person is empty) that the customer has replied.  The customer is able to update the Additional Information field directly and any email reply from the customer is mapped to that same field.

The closing email to customers includes a hotlink back to the record if they wish to reopen it and instructs them to explain why they are not satisfied with the solution.  Clicking the hotlink will

automatically change the "I Would Like To Reopen My Incident" field to Yes, which in turn sets the Status of the Incident to Reopened and notifies the assigned person (Rule: Incident Customer Update Actions, Action: Submitter Update Actions).
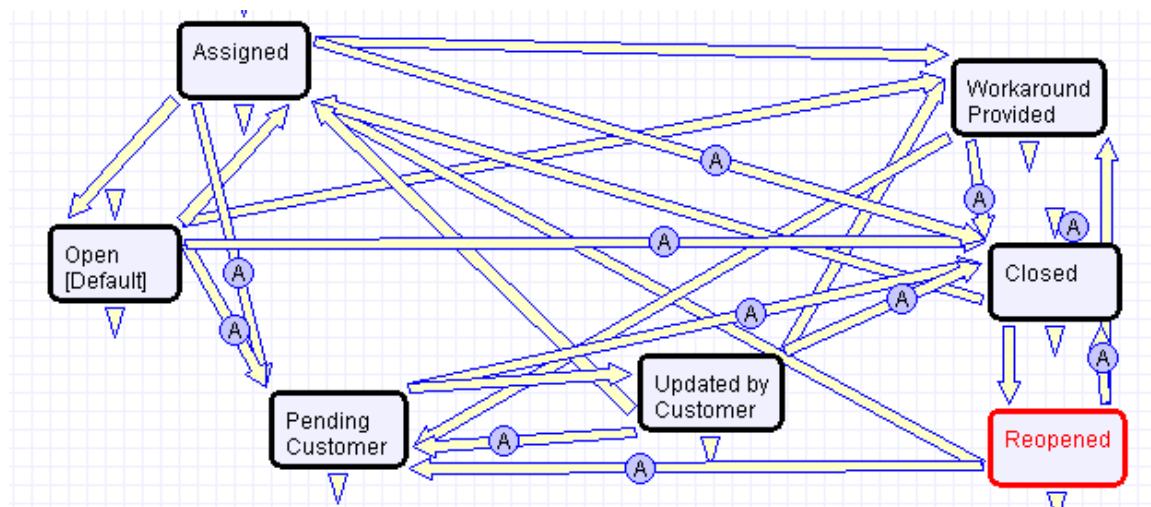
## WORKFLOW



*Incident table default Workflow*

## FIELDS

Fields are shown in the Incident Fields table in the Appendix.

## OWNERSHIP

Records in the Incident table are "owned" by the individual submitter. This means each record is associated with a particular employee login and no other "end user" employee will be able to edit that record. All technician users are able to edit any Incident by default.

## REPORTING AND STATISTICS

Several fields are included that are used for statistical reporting. The following fields may be of interest:

Number of Assignees – is auto-incremented each time the assigned person changes (Rule: Incident Edit Actions, Action: Notification Actions)

Number of Teams Assigned – is incremented each time a different team is put in the Assigned Team field (Rule: Incident Edit Actions, Action: Notification Actions)

Number of Reopens – is incremented each time a closed request is reopened by a customer

Total Hours to Close – elapsed time between Date Created and Date Closed

Working Hours to Close – elapsed time between Date Created and Date Closed minus the non-working hours of the team in the Assigned Team field and the time during which the Status was Pending Customer

There are default reports measuring Total Time Spent by Type of Problem, Average time to close by Type of Problem and by person who closed, as well as averages for number of people assigned, number of teams assigned, number of reopens, and so on.  With the field structures already there, it is easy to add reports to slice and dice the information the way you need it.


# PROBLEMS TABLE

## OVERVIEW

Problems represent the root cause of one or more Incidents or possible Incidents. Resolving a problem means resolving/preventing related Incidents.

Incident Management is concerned with restoring service as quickly as possible, whereas Problem Management is concerned with determining and eliminating root causes (and hence eliminating repeat problems).

The primary objectives of Problem Management are to prevent Incidents from happening and to minimize the impact of Incidents that cannot be prevented.

## USE CASE

### Problem Creation

When an incident is determined to be based on an underlying Problem, first-response support technicians create a Problem record from the Incident, automatically creating a link between the

Incident record and the Problem. The new problem record imports relevant information from the Incident, such as the linked Configuration Item. Problems are also creatable independent of existing Incidents, such as in cases where a problem is discovered internally but no Incidents have been reported.

Resolution of problems may require Changes to the system. Staff addressing the problem may determine that a shared CI needs to be replaced or modified, and may therefore file a Change Request.

When the problem is resolved, a technician updates the record with relevant information and closes the record, in turn prompting automation to begin closing procedures for related Incidents. If the problem cannot be resolved it may be classified as a Known Error and a permanent work-around supplied. This will also update the related Incidents.

## Processing of Records

Once created, Problem records can be linked to Incidents from either the Problem record or from an Incident record.

Priority, a measure of the Problem's urgency and relative importance, is set by default to the Priority of the spawning Incident, but can be changed at the time of creation. Problem Priority, a measure of impact and risk, particularly with regards to IT service operations, may be determined separately as part of Problem Classification procedures. Together Priority and Problem Priority help IT managers make factual decisions for scheduling and planning and help determine the category of related Change Requests. By default new Problems have a Problem Priority of Standard (0), indicating no special review or authorization is needed.

## Diagnosis

The Problem follows its own workflow separate from the Incident. Ops team members open the problem record in the default state of "Pending Diagnosis" to indicate that a diagnosis has yet to be determined, or "Diagnosed" if no further steps are required. A record may sit in a state of Pending Diagnosis for some time before staff actually begin to perform the diagnosis, so Start Clock and Stop Clock buttons let users indicate how much time was spent diagnosing a particular problem.

As part of the diagnosis, technicians select the service involved based on existing Incident reports or based on the most likely service to be affected by the Problem. If a particular Configuration Item is identified as the source of the problem's root cause, staff can quickly link the Problem to the CI record.

Once a diagnosis is supplied, staff will move the record into a state of "Diagnosed" and fill out the Root Cause description. They may suggest a temporary fix, or "workaround", for the problem and related incidents (e.g. "use Printer B3 for now instead") if a permanent solution is not readily available. While determining a workaround and/or permanent resolution, technicians can use Start and Stop Clock buttons to track the time spent determining workarounds and solutions for this Problem.

If at any time during the root cause diagnosis or determination of the proper solution staff need more information from a separate process, such as Incident details from first level support, the Problem record may be placed in a state of "Pending More Information".

If a Problem is deemed too risky or of lower priority than more imminent issues, it may be put in a status of "Deferred" to reflect no ongoing diagnosis or pending changes.

## Solution

In most cases where a Problem's root cause deals with a Configuration Item, a Change Request will be submitted to make the appropriate fixes to the Configuration Item. Change Requests are creatable directly from the Problem record, instantly linking them. While a problem is waiting for a Change Request it can be put in the status of "Pending Change".

If the Change Request linked to a Problem is closed, the system will send an email notification to the problem assignee so that the individual can take additional steps to close the Problem record if it was pending change for resolution.

A problem whose root cause is known but for which there is no permanent resolution is considered a Known Error. Known Errors should have Workarounds to allow Incident Management to restore service as quickly as possible. The "Update Incident with Workaround" button allows staff working the Problem to quickly disseminate workaround information to linked Incidents with the click of a button. Clicking the button will post the workaround in all related incidents and change their status to Workaround Provided, which will also trigger an email to both the end user and the assigned staff person of the Incident(s).

A similar button is used to transfer the problem Solution to related Incidents. Clicking the "Update Incidents with Solution" button in the problem will populate the Solution field of the Problem into the Incident Solution field and set the status of the Incident to Closed, emailing the customer.

Once a permanent resolution is determined and implemented, staff users enter the description in the Resolution field and set the status to Resolved. If the resolution contains information that is useful outside of this problem's particular scope, the "Add to Knowledgebase?" field can be set to Yes to make the Resolution field available via FAQs.

## Ownership

Problems are "owned" by the staff member who creates the Problem record. Since only internal staff will see Problem records, groups may share responsibilities between Incident Management and Problem Management and multiple individuals may share ownership over time.

## Workflow
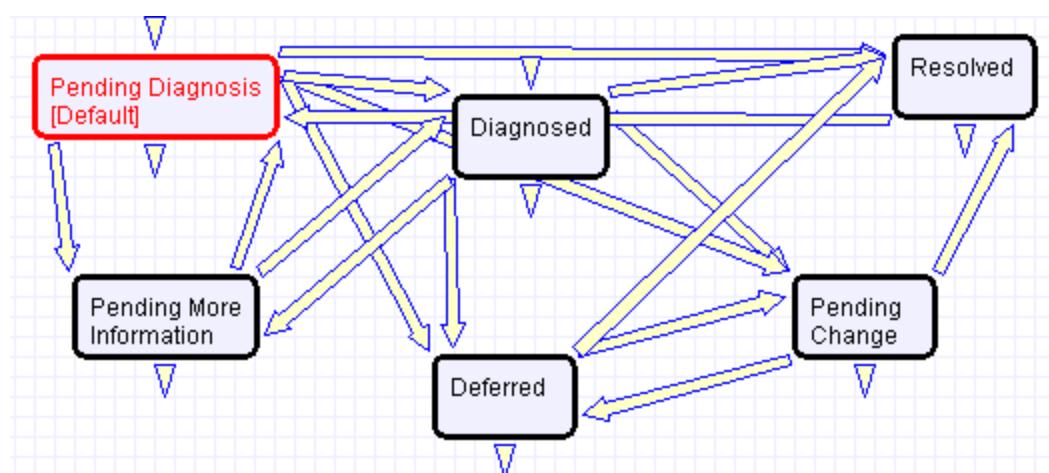


## PROBLEM FIELDS

Please see Problem Fields in the Appendix.

# CHANGE REQUESTS (RFC) TABLE

## OVERVIEW

The Change Requests table is used for Change Management. A Change request is created when a change is needed to a configuration item or to any other business object that may require a set of approvals before such a change can be completed.

Once a change request is created, it can be assigned to the appropriate teams or individuals for one or many approvals, and may then be moved along in the process, from approval to scheduling, to action by staff members actually making the change, to completion, testing, and release.

Of course each organization has its own definition of types of change that may need to be managed in this way, but we have prepopulated the service catalog with a list of common change types. You can easily add your own change categories and types to the services table.

Change Requests differ from Service Requests in that they are not visible by default to end users and typically follow a stricter set of procedures and approvals.

The Change Request table pulls its services from linked entries in the Services table. A user creating a change request chooses the category of service from a drop-down, and then sees all the active services that belong to that category in a second drop-down list called Service Title. The default service categories for change requests can be seen in the Configuration Items table (or by creating a CR record and opening the drop-down list). When the service is selected, several additional fields are pulled in with the service, including the number of approvals needed.

# MANAGING APPROVALS

There are many different ways of structuring a multiple approval process, and the best method depends on your company's particular circumstances.

We have tried to make the out-of-the-box implementation as simple as possible by creating hard coded approval fields ($1^{st}$ approver, $2^{nd}$ approver, $3^{rd}$ approver and so on) with up to 5 separate approval teams/persons/approval dates/yes-no approval fields for each change request. If your change requests typically require 3 or fewer approvals, this is probably the simplest setup to manage, and we have already preconfigured some essential approval notifications and validations for you.

If your changes often require more than 4 approvers or approval teams, this is probably **not** the ideal setup for you, and an embedded table of approvals would likely make more sense. It is easy to add a table called Approvals and embed that table in the CR table so that when a CR request is created, an unlimited number of related approval records can be created, each with their own status, assignee, access, automatic notification, and so on. Automating approval behavior with an embedded table is a bit more complex than with hard coded fields, particularly if you would like to automatically create a set of approval records for specific change services or based on some other criteria. However, such automation can certainly be done, and our professional service team would be happy to assist you in meeting your precise needs.

The current setup shows from 1 to 5 sets of approval fields based on the service selected and the value in its "Number of Approvals Needed" field. It allows the technician to identify the appropriate approval teams or individuals and does not automate this. It also allows the number of approvals field to be modified by anyone with the appropriate permission to expose or hide additional approval fields.

Details of the built-in approval automation are provided below in the Use Case section and in the Change Request Rules section of the Appendix.

## PARENT-CHILD CHANGE REQUEST STRUCTURE

There are a few different structures that can support changes that involve multiple CI's, such as upgrades, reconfigurations, replacement of hardware and so on.  In order to maintain a history of all changes to each CI, there must be an individual record created for each task done on a CI.  One method is to use an embedded Tasks table within the Change Request table for creation of tasks related to individual CI's.

The other method is to use a parent-child CR structure, with Child CR's used to handle groups of changes to CI's under a single approval.  The advantage of this method is that all work on a CI that would be covered under a Change Request is shown in a single embedded table in the CI record.

The system is designed so that for a major upgrade that involves multiple CI's, a parent CR will be created, and then child CR's can be auto-created for each CI that is to be changed.  The approval process is only necessary for the Parent CR, and when the parent CR's Status is changed to Approved, all Child CRs of that parent are automatically set to Approved so that work can begin. If the Parent CR Status is set to Rejected, the Child CR's are likewise automatically set to the same Status.

This method is based on two principal fields: the Related to field and the CR Type field.
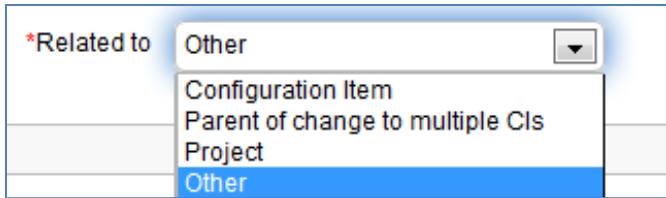
"Related to" has possible values of Configuration Item, Parent of change to multiple CIs, Project, and Other.  The CR Type field has possible values of Independent, Parent, and Child, with Independent the default value.

If the Related to field is set to "Parent of change to multiple CI's", then the CR Type field is automatically set to Parent, and a multi-value linked field called "CI's for Child CR's is made visible.  This allows the user to look up multiple CI's and import them into that field to indicate that a child CR should be created for each one.  When the record is saved and that field has changed, a Child CR is automatically created for each CI, copying most of the main fields from the parent CR, but prefixing the Change Summary with the words: "Child of" (the parent change summary).

These Child CR's, once created, are then displayed in the embedded table within the Parent for tracking of their progress.  These new requests are automatically given a CR Type of Child and a Related to value of Configuration Item.

If additional CI's need to be added to the parent CR after some CI's have already been added to that field, the user can use the lookup to find the additional CI's and choose the option to Import and Replace to overwrite what was in the field before, then save the parent CR again.  This will generate new child CR's for the newly entered CI's.

Each Child CR is assigned according to the normal assignment logic (see below).

## USE CASE

Change Requests (CRs) may be directly created by technician users or may be created by conversion from a Service Request or a Problem, in which case they are linked back to the generating record.

### End User Record Submission

By default, Change Requests are not visible to end-users and cannot be created by them.  This is defined by group permissions, and if you would like to allow end users to submit change requests, you can simply change the group permissions of the relevant groups to enable this.

### Technician Record Submission

When a technician submits a Change Request, the Requestor contact information fields automatically populate based on the details in his/her User record. In the default setup, a request cannot be submitted on behalf of a user who has no user record in the system.  However, the creator can select a different person as the requestor if needed.

The user is required to select a service after selecting a service category. When the service is selected, the user will see a description of the service, and any special instructions for that service. The service will also define the default number of approvals needed. Users in the appropriate groups may be given permission to change the number of approvals needed for a particular change.

When a technician creates the CR he may choose an assigned team and assigned person from the drop-down fields on the Work Notes tab. If he does not choose the team, when the record is saved, the assigned team will be auto-populated based on the logic described below.

New CRs are created in the Status of Open by default. During creation, a technician may change the Status to Pending Approval to reflect that the Change Request is ready to begin the approval process.

## Automatic Assignment of New Requests

When a new Change Request is saved, it is automatically assigned to a team based on the values in three fields pulled from other tables (Rule: CR- All Creation Actions, Action: Assign Change Request). The related service has a Responsible Team associated with it and a field called "Give Service Team Priority over CI Team" with a Yes/No value. If a Configuration Item was identified for the request, then there is also a CI Responsible Team field that will be populated in the CR.

The current assignment logic is: if the Assigned Team is empty when created and the Change Request involves a specific configuration item (CI), then the request is assigned to the responsible team for that CI, unless the "Give Service Team Priority over CI Team" is set to Yes. In this case, or If no CI is defined, then it is assigned to the Responsible team for the Service. If there is no Responsible Team for the Service, then it is assigned to the Change Management Team.

Naturally, this logic may be changed to suit your organization. You can simply set a default value such as the Change Management Team or you may choose some other logic to use based on different fields. The current logic is implemented in the CR rule named "CR: All Creation Actions" and the action named "Assign change request" and may be modified there.

## Automatic Emails Sent upon Submission

If the status is not Closed when saved, the rule named "CR- All Creation Actions" will send the requestor an acknowledgement email and will also send an email to either the Assigned Person, if there is one, or to the Assigned Team.

## Processing of Records

Most services linked to Change Requests require one or more approvals, as defined in the Service record. If the technician chooses approver teams for the approvals on the Approvals tab and changes the status of the Change Request to Pending Approval, a rule action will automatically send an email to the approver(s) or approval team members requesting their attention (Rule: CR Approvals, Action: I: Approval Notification Actions). The email includes a hyperlink for the user to click to edit the Change Request directly.

The same rule requires that anyone editing the approval fields be a member of the team indicated in the Approval team field (Rule: CR - Approvals, Action: Approver Validations).

Note that Child CR's do not require approvals – these fields are ignored and the status is automatically changed to Approved when the Parent CR is approved.

If a technician needs to reassign the Change Request to a different team or person, he or she simply changes the Assigned Team and/or Assigned Person field and the system will email the new assignee notifying them of the reassignment (Rule: CR - Edit Actions (API Enabled), Action: Assignee Change Notifications).

Once all necessary approvals have been received, a technician can change the CR to a status of Approved (Rule: CR - Approvals, Action: Status Change validations**).** When a CR changes to either Approved or Rejected the rule will send emails to the Assigned Person or Team.

When technicians have finished work on the Change Request they set the Status field to Closed. This triggers a workflow email to the requestor telling them that the request has been completed.

## Parent/Child Visibility

Change Requests are independent of other Change Requests by default. However, if the CR Type is set to Parent or Child, the Related Change Request(s) section shows details regarding the Parent/Child relationship of a given CR. The CR Type field is always visible and shows the CR type of the current CR: Independent, Child, or Parent. The options for this field change based on the value given in the Related to field.

If CR Type is Parent, the Related Change Request(s) section will show a related table of linked Child Change Requests and an action button labeled Spawn Child CR, which will create a new CR linked to the current record and put it in the related table. If the CR Type is Child, the Related Change Request(s) section will show a set of linked fields to a single Parent CR record. No additional fields are shown if CR Type is set to Independent.

## Reporting Time Spent

Technician users may easily report the time they spend on handling change requests. There are two fields and an action button: Time Spent, Time Description, and Add Time on the Work Notes tab of the layout. Entering values there will automatically create a new time entry record when the Add Time button is clicked. The time entry will show the work done by the technician and on the current date.

All the time entries for a request can be seen on the Time tab as well. If a technician needs to report time that was spent on a different day or by a different user, he may click the New button on the Time spent table to submit a time entry directly and change the date or "Done by" field. All time entered is totaled in the All Time Spent field, which can be used in reporting or billing.

Note that this same time entry methodology is used in the Incident, Problem, Service Request, and Task tables.

## WORKFLOW
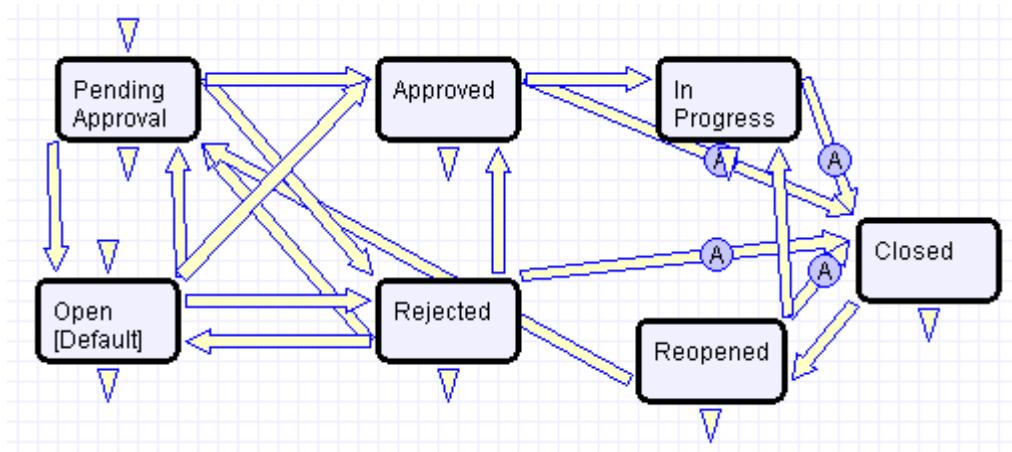
## FIELDS

See [Change Request Fields](#) in the Appendix.

## OWNERSHIP

Records in the Change Request table are "owned" by the person whose login matches the Creator Login field.

## REPORTING AND STATISTICS

### Forward Schedule of Change

The Forward Schedule of Change represents approved changes and the proposed implementation dates. More generally, it is a list of Change Requests set to occur in the future. Functionally it might include CRs with restricted criteria, such as approved changes only, but by default the Forward Schedule of Change simply shows all Change Request that are not in a status of Closed.

The Forward Schedule of Change is a saved search available from the vertical navigation menu under the Change Request table name. Change Requests are listed in order of Requested Date of Completion. This search is also used in a structured HTML report of the same name. The report groups the CRs by requested date of completion (by week) and shows summary data, including the sum of the Estimated Time to Complete for all records. The report can be configured to be sent in a weekly email to the Change Management team.

### Accomplished Change

Accomplished Change is the complement to the Forward Schedule of Change and by default shows all closed Change Requests, sorted by Date Closed.

## CONFIGURATION ITEMS TABLE

The Configuration Items table holds records containing information about your company's Configuration Items (assets).  It may include CI's that are in inventory and not in service, as well as all assets that are in service.

## USE CASE

### Configuration Item Creation

Staff may add a Configuration Item manually to the system by creating a new record, or integrated Discovery systems may create CI records automatically with specified attributes.

Configuration Items have type-specific attribute fields (Manufacturer, Model, Serial Number, etc.) and relationships to other CIs, Service Requests, Problems, and Change Requests.

Related Service Requests, Problems, and Change Requests are displayed as lists of records within a CI, so the history of a given CI is readily available.

Configuration Items may also be related to other CI's in an upstream and downstream embedded table.

### Record Creation

Configuration Items may be created independently or when selecting Configuration Items in Change Request and Contract records. Configuration Item Name and Configuration Item Type are required fields.

### Ownership

Configuration Item ownership is defined as the user whose Login matches the User Login field in the Configuration Item record.

### Processing of Records

Configuration Items follow a lifecycle workflow from ordering through retirement. There are no business rules associated with the Configuration Item table.

Configuration Items are created in the state of Installed by default (see Workflow diagram below). Configuration Items that are requested but not immediately available might be created in a state of On Order, moved to In Stock when the Configuration Item is received, Pending Install while Operations is tasked with installing the Configuration Item, and finally Installed. Installed Configuration Items can change to In Maintenance for repairs and Retired or Stolen when the Configuration Item is no longer in use.

## WORKFLOW
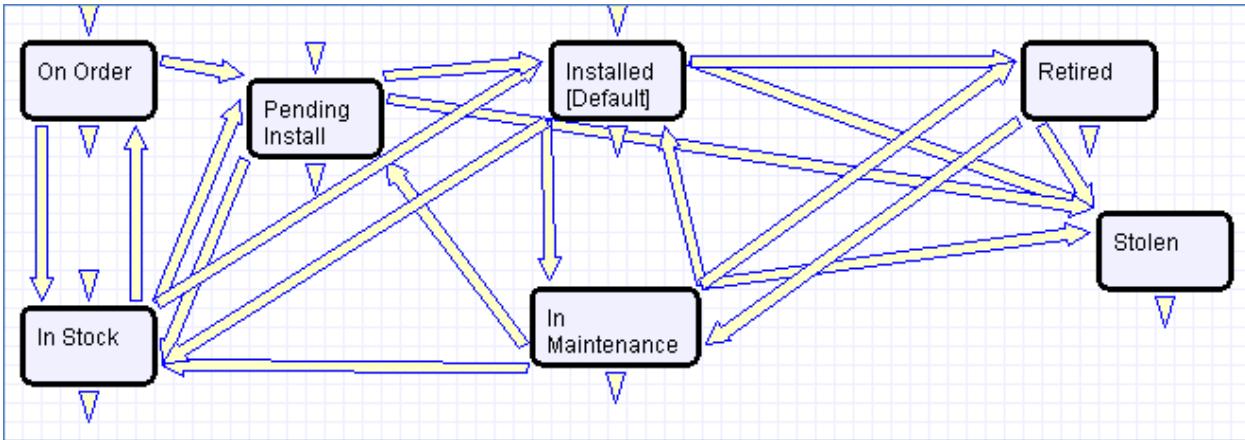
## FIELDS

The Configuration Item fields are described in the Configuration Item Fields Table in the Appendix.

# PURCHASE REQUESTS TABLE

## OVERVIEW

Purchase Requests are used to enable an end user to request new equipment, software, or furniture. A single purchase request can include multiple requested items, which are selected from a list of available items.

## EXAMPLE USE CASE

An employee end-user (*lsmith*) needs a new laptop. She logs in and creates a new Purchase Request from the End-User interface. She'll specify who it's for (the record's creator is the requestor by default), the type of purchase by Service Category, the business justification for the request, etc. Lastly she'll select the item(s) she's requesting via a Related Table to the Items Requested table, which in turn draws from the Items table. She may also select from Configuration Items in stock that matches the type of purchase she chose.

Jim Sullivan (*jsullivan*), *lsmith's* manager, receives an email that a new request needs his approval. He can approve or reject directly from hotlinks in the email or edit the record manually from the staff interface. Jim, as a supervisor, will change the Supervisor Approved field to either Yes or No. A rule running on edits will then change the status appropriately (to Supervisor Approved or Supervisor Rejected). If he approves, a notification will be sent to the Procurement Team.

Members of the Procurement Team (e.g. *awilliams*) receive an email stating that a new Purchase Request is awaiting their approval. Members of the Procurement Group will see some additional fields in the Purchase Request record, including a Procurement Group Approved field similar in function to the Supervisor Approved field. They will also see another Yes/No choice field called "All Items Received?" When the recipient/requestor has been satisfied, the Procurement Group changes the status to Fulfilled.

## Relationship Diagram

41

```
              ┌──────────────────────────────────────┐
              │          Requesting Users            │
              └──────────────────────────────────────┘
                    ╱                          ╲
      ┌──────────────────────┐      ┌──────────────────────┐
      │   Purchase Request   │      │   Purchase Request   │
      └──────────────────────┘      └──────────────────────┘
        ╱              ╲                        │
  ┌──────────┐   ┌──────────────┐      ┌──────────────┐
  │   Item   │   │ Item Requested│     │ Item Requested│
  └──────────┘   └──────────────┘      └──────────────┘
        │              ╲                  ╱
  ┌──────────┐       ┌──────────────────┐
  │   Item   │       │       Item       │
  └──────────┘       └──────────────────┘
```
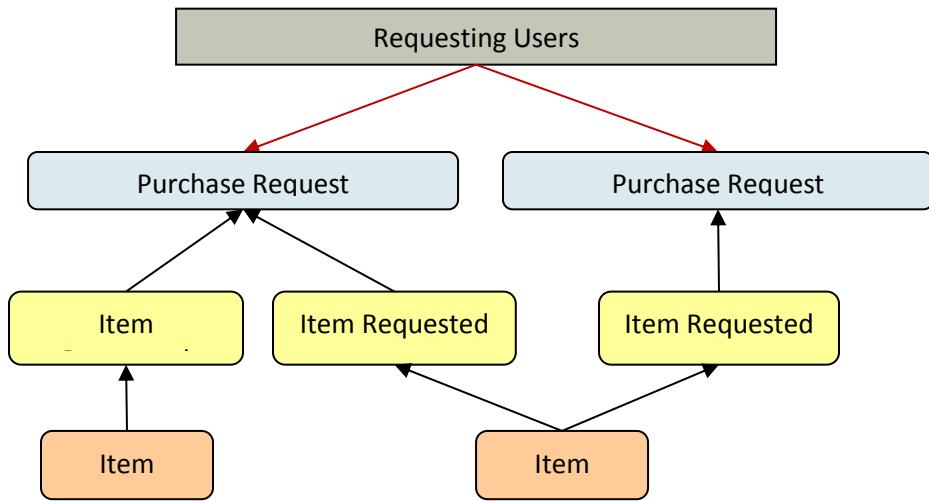
## Record Creation

 Purchase Requests are creatable by internal customers from the End-User Interface, or by members of the Purchasing team on behalf of other users.

Upon creation, a rule sends emails to the Requestor, the Requestor's Supervisor, and the Purchasing Team notifying them of the new Purchase Request.

## Processing of Records

Once created, Purchase Requests follow a two stage approval process.  The creation emails sent to the Requestor's Manager and the Purchasing Team lets them know that their approval is required and provides a link in the email to edit the record. A supervisor can edit the "Supervisor Approved?" field to yes or no, and the Purchasing Team edits the "Procurement Group Approved" field to yes or no. Rules validate these changes and move the Purchase Request to the appropriate status, such as "Supervisor Approved" when the supervisor changes the field to Yes.

Once fully approved, Purchase Requests are Ordered, Received, and Fulfilled. Ordered and Received are used if the requested items are not immediately available, and changing the "All Items Received?" field to Yes will move the request into the status of Received. Purchase Requests can move to a status of Fulfilled any time after approval.

## WORKFLOW
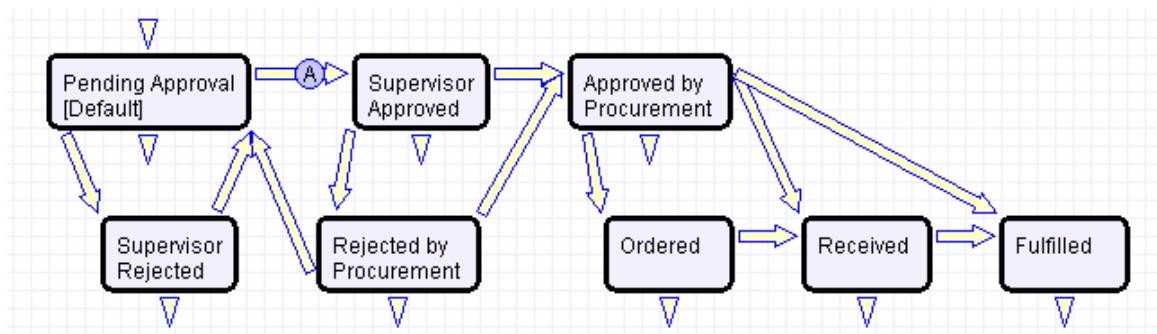
Purchase Requests have a two stage approval process, first by the Supervisor and secondly by Procurement staff. Once approved, Purchase Requests can move to Ordered/Received states or directly to Fulfilled.

## OWNERSHIP

Purchase Requests are owned by the user whose Login matches the Creator Login field in the Purchase Request.

## FIELDS

See Appendix A: Purchase Request Table fields

# ITEMS REQUESTED TABLE

## USE CASE

The Items Requested table acts as an intermediary table for the Purchase Request and Item tables. When users request items as part of a Purchase Request, they are creating new Item Requested records. These records represent the Items selected and their quantity for a particular Purchase Request.

### Relationship Diagram
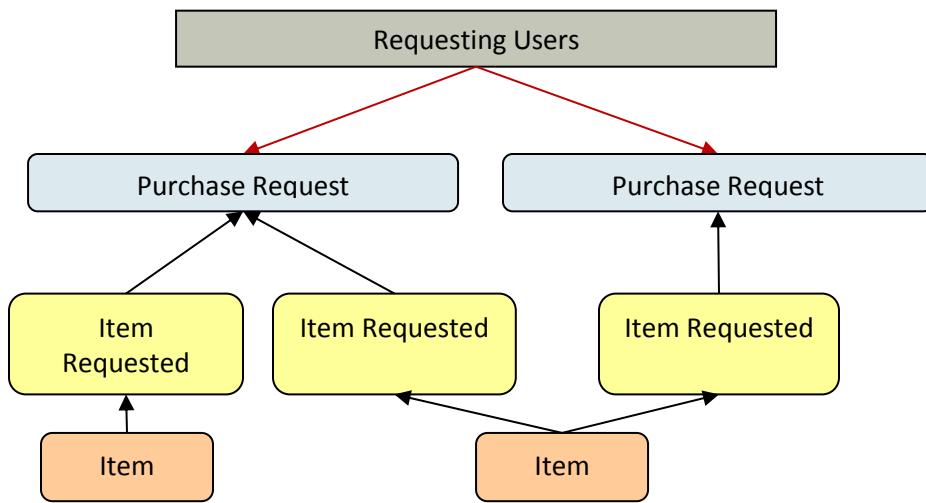
## Workflow

These records have no workflow and instead follow the Purchase Request record they are linked to.

## Record Creation

From a Purchase Request record (see Purchase Request Use Case), a user creates new Item Requested records for each Item desired in the embedded table view. After selecting the Type of Purchase in the Purchase Request record, clicking "New" from the Purchase Request linked Items related table presents the user with the new Item Requested form, filtered to items that fall under the selected Purchase Request Type of Purchase.

The Item Requested form has three major sections: Item Requested details, linked Purchase Request Information, and an area to select a matching Configuration Item. The requestor selects an item from the Item Name drop-down box, which is filtered to show only items in the Type of Purchase category chosen in the Purchase Request record.

The Purchase Request Information fields are automatically populated with data from the Purchase Request this Item Requested was created from and represents the link to the Purchase Request.

The Configuration Item section allows the user or fulfilling staffer to select an existing Configuration Item from the CI table. The CIs available for selection are filtered to only in-stock Configuration Items that match the Type of Item for this Item Requested. The CI fields link the Item Requested to a physical, tracked asset (CI) rather than just the template description (Item), enabling greater control and accurate recordkeeping.

Clicking the "Assign to Requestor" button will automatically update the selected Configuration Item by linking that CI's "User" fields to the same Contact linked here as the Requestor. That is, the button will assign the CI selected to the person requesting the item.

## Processing of Records

Once created, Item Requested records remain unchanged and stay linked to the Purchase Request. Since the Item Requested records are maintained separately from the Item records they were spawned from, they retain data that might change over time, such as Unit Price. Thus, even when Items change, Purchase Requests will always show Item data as it was at the time of the Purchase Request and reflect an accurate total.

## Ownership

Item Requested records are "owned" by the Contact record whose Login matches the Creator Login field in the Item Requested. Since Items Requested are not maintained outside of a Purchase Request, the relevant ownership stems from the Purchase Request record.

## FIELDS

See Appendix A: Item Requested Table Fields

## DEFAULT LAYOUT



## ITEMS TABLE

## USE CASE

The Item table stores information for each type of item that can be requested as part of a Purchase Request. The Item Requested table allows the user to select from the Item records in this table when creating a Purchase Request.

### Record Creation

Item records are created by internal procurement staff users to represent items that may be selected by end-users for Purchase Requests.

For example, say the Dell Inspiron 2000 Desktop PC is the default model for IT staffers and all employees are set up with one when hired. Procurement staff in charge of maintaining Purchase Requests would create a single record in this Item table to represent the Dell Inspiron 2000 model with all relevant details, such as Make/Model, Unit Price, and descriptive text/choice fields. When users request a Dell Inspiron 2000, an Item Requested record is created from the Item record as a template and then linked to the Purchase Request.

### Processing of Records

Once created, Item records require little modification. By default, new Items are created as Active and not requiring Supervisor Approval. If a particular Item type is unavailable, no longer in use, or not ready for distribution, Procurement staff may set the Item to Inactive. Expensive or controlled Items can be toggled to require Supervisor Approval, which can act as a flag to apply approval processes when the Item is requested.

## WORKFLOW

Items are not associated with any status fields, but instead contain an Active/Inactive choice field to differentiate selectable Items from non-selectable Items. By default, inactive items are not available for Purchase Requests.

The Item table does not have a Workflow State diagram.

## OWNERSHIP

Items are owned by the user with the login in the Creator Login field. However, since Items are templates for Requested Items, ownership is less relevant to the individual and records will likely only be maintained by a Procurement Group or similar staff team.

## FIELDS

See Appendix A: Item Table Fields

# BACKGROUND TABLES

There are several other tables essential to the system that holds data that is less active and workflow-oriented than the request tables.  These tables are briefly described here.

## COMPANIES TABLE

Purpose: This table holds information about companies that interact with your organization.  It may include customers, vendors, prospects, manufacturer, and so on.

### USE CASE

Companies may be created by conversion from the Leads table. Members of the Admin, Professional Services, Support Staff and Sales groups may also create new Company records directly.

The Company field contains mostly static data, and thus does not have any associated workflow actions.

Information about a company and a company's relationship to your organization is stored in a Company record.  This information can be everything from the company's billing address to information about their support contracts with your company.  More complex data relationships are displayed via linked fields, which pull any records from the Contacts and Opportunities tables that reference the Company record.  This allows you to see any business deals or transactions in progress and keep track of any Company employees that your organization contacts.

#### Ownership

Records in this table are "owned" by the individual assigned sales rep, so each record is associated with a particular user login.  Admins and members of the Contract Owner, Contract Manager, Marketing, Project Manager, and Sales groups can view and edit Companies. Most groups can view their own Company, and most internal users can view others' Companies.

## PEOPLE TABLE, EMPLOYEES AND END USERS SUB-TABLES

Purpose: The User table tree is used to store information about individuals who use your system.

### USE CASE

The People table contains data about individual users of your system and individuals with whom your organization has contact or business relationships.  There are two basic subtypes of Users: End Users and Employees.  Most external Contacts will go in the End User table, as the Employee table tracks information that does not apply to people outside your company, such as working hours and hire date. People records normally have a link to a record in the Company table to define which Company is related to the Person. By default, People records have an embedded list of records from another table, called a related table, listing all of their Support or Helpdesk Case records (tickets).

People are considered static in that they do not change status.  By default, no workflow is specified for the People table.

People may be created manually by Guests, Support Staff, and members of the Professional Services, Sales staff and Admin groups.

Guest record creation is enabled by default so users can create their own logins using the limited-access "register" account.  Records created by the Guest login "register" are added to the "Selfregistered users" group automatically by a rule.

People may also be created as the result of a conversion from a Lead record, or may be created as part of an import from an external database such as an LDAP directory.

Each Person record is assigned either a Team or an individual from the Employees subtable as the "Owner".



The Employee sub-table has a rule attached to it that runs a conversion of an Employee record to the Approvals table when triggered by a field changed on the Employee record. This facilitates the Documents approval process.

## CALENDARS TABLE

Purpose: This table contains event records displayed in the Calendar block in the left pane, or using a normal table view.

### Special Use Case

The Calendar table is a special table used in the Staff interface Calendar pane feature. Records in this table display as events in the Calendars for the individuals and groups that they reference. Calendar

records are mostly static and do not have any default associated workflow, rules, email setup, saved searches or charts.

The Calendar pane is only available to Staff users, and cannot be made available to End Users.

By default, only members of the Admin, Professional Services and Support Staff groups may create Calendar records, and Professional Services and Support Staff may only view their own. No other groups have access to view Calendar entries by default, so other staff groups using the Calendar pane must be individually activated or given view and edit permissions.

Records in this table are "owned" by their creators, and are linked to the creator's login.

## EUI TEMPLATES TABLE

Purpose: This table contains HTML files used in creating a customized End User portal. It has sample files that can be used, and you may create your own html files and upload the html content to records in this table to serve up the pages.

### Special Use Case

The EUI Template table is a special table used for storing HTML files used in changing the presentation of the Agiloft end user interface. Since these files undergo little or no change during their lifetimes, no workflows, rules, or actions are associated with them, and no charts, reports or saved searches exist for this table.

### EUI and Documents Table

When the Documents Table is enabled, the EUI can be configured to allow access by end users to the table. In order to enable this, the "main.php" and "menu.htm" files need to be edited so that all references to "global.menu.file" and "global.home.file" are "globoal.menu-doc.file" and "global.home-doc.file", respectively. These references can be found in '#ew_include("$ewText.get('')")' calls in those html pages.

## ADDITIONAL FUNCTIONS

In addition to the tables and functions described above, several additional functional tables are included with the ITIL KB that may be activated for use depending on the functions you requested when signing up for a free KB or downloading and installing. They cover the general functions of Contract Management, Project Management, and CRM (customer support, sales, and marketing). These include the following main tables in addition to their supporting tables: Contracts, Documents, Projects, PO's, Surveys and Survey Answers, Leads, Opportunities, Quotes, Products, and Campaigns.

## APPROVALS TABLE AND APPROVAL TEMPLATES SUBTABLE

Purpose: The Approvals Table is used to hold all of the approvals that are processed through the system. Each record in the table is an individual approval or rejection that is linked to a parent Contract. Approvals may also be linked to Documents and Change Requests.

The Approval Templates Table is used in the generation of required approval records for Parallel workflows. For more information on Workflows, see the section titled: Workflows Table.

## USE CASE

All approval records are created by the system as a result of actions taken in the associated Contract or Document record. Approvals whose source is a parallel workflow are generated from an Approval Template record, detailed later in this section. Approvals whose source is a sequential workflow are generated through conversion from within the Contract record. For more information on Workflows, see the section called Workflows Table.

### Approval Records for Documents

All approval records for document management are created as a result of a conversion of an Approver Employee record with the Approver mapped from the Full Name field of the Employee record.  The Approval record will have information from the linked Document record along with a link to the attached documentation.

The Source field will be updated to include a value for Document management.  On the creation of an approval record with a source of Document management, the Approval Team field will be updated to reflect the Document Reviewers team and the reviewer is automatically notified that they have a document to review and approve.

The linked set from the Document record in the Approval record will include a hyperlink to the document using the "view only source field display" for the attachment field. The reviewer can click the link to launch the document – allowing the reviewer to mark up the document which can then be uploaded to the approval record (reviewers will not be able to upload the document to the source record).

## Other Approval Records

Each Approval record stores the parent record ID, the Approval Team, and the user who submitted the approval. The timestamp of the approval is also captured and displayed.

Notes regarding the approval or rejection are entered into the **Approval Notes** field and are displayed under the **Approval Running Notes** section after being saved.

For information on how approval records are generation, refer to the *Contracts Table: Staff Use Case* section above.

## Approval Template Records

Approval templates may be created by users in the Admin and Contract Management groups. Templates are used exclusively for parallel workflows.

Each Approval Template contains information about which table the template is related to as well as the Approval Team. The Approval Status field is manually set by the approver in parallel workflows; it is automatically updated through an Action Button in sequential workflows. The Approver field is used by the system in the triggered conversion.

Each Approval Template may be used in multiple parallel workflows within a table. The Workflows tab displays all the workflows in which the current approval template is being used.



## OWNERSHIP

Approval records and Approval Template records in this table are owned by the user that creates them. Specifically, a record is owned by the user whose Login matches the Creator Login field.

## WORKFLOW

The Approvals Table and Approval Templates subtable have minimal workflow associated, and thus state changes are handled through the rules automation rather than the graphical workflow interface.

## REPORTS

The Approvals table and Approval Templates sub table do not currently have any Charts/Reports set up.

# CONTRACT MANAGEMENT TABLES

The Contract Management system is designed to be ready out of the box with a variety of approval processes built-in.  It is easy to eliminate or modify any functions to better fit a desired business process.

Some time-based rules are disabled by default and need to be enabled for the table to work correctly. To do this, go to the Rules tab of the Setup Contracts page, edit a rule with "(Disabled)" in the title, and click "Yes" in the "Rule is enabled" section at the bottom of the page.

This section describes how the default contract management system is configured.

The Contract Management system is comprised of a few main tables with background tables playing a supporting role. The main tables are described below.

## CONTRACTS TABLE

Purpose: The Contracts table holds all contract records. It also controls all associated automation and notifications related to contracts. A representative record can be seen below.

A contract can be related to one or more Configuration Items and is generally related to a company record, for either a vendor or customer. Each contract also has an owner at your company who is notified when it is time to renew the contract.

Required fields are marked with a red asterisk.

## Staff Use Case

This section covers the use case for staff members inside Agiloft.

Each record in the table holds information about an individual contract including Vendor details, Insurance Certificate expirations, Approval information, and Renewal details.

### Contract Creation

Contract records may be created by members of the Admin, Contract Manager, Contract Owner, Project Manager, and Sales groups.

Contracts may be created in one of two ways:

- Hitting "New" in the Contract Table action bar

- Using the **Create Related Contract** button in the Related Contracts/Renewal tab to create a renewal, subcontract, or amendment from the current contract. This button is available only if the contract is not in a Status of Draft, Pending Contract Manager, or Pending Approval.

Creating a new contract the second way will automatically link it to the current contract by populating the Parent Contract ID field. A more detailed explanation of creation by this method is handled in the *Related Contracts/Renewal* section.

Contracts fall into one of four categories:

- Contracts

- Master Contracts

- Subcontracts

- Amendments

The category is indicated by the **Record Type** field in the contract Common Area. "Contract" is the default for newly created contracts. It can be used to indicate either a stand-alone contract or a contract that exists under a Master Agreement. In the latter case, the Parent Contract ID should be filled in. Subcontracts and amendments should be linked to a Parent contract.



Below the **Record Type** are fields storing the Assigned Team and the Internal Contract Owner. The default **Assigned Team** is the Contract Management Team. This field is changed automatically by the system during the Approvals process. The default **Internal Contract Owner** is the user who creates the contract record.  The list of available choices is filtered to users who are in the Contract Owner Team or the Contract Management Team. Users in the Admin Group may manually change the **Assigned Team** and **Internal Contract Owner** if needed.

All contracts are created with a default Status of Draft. The contract Status is changed automatically by the system at appropriate places in the workflow; however users in the Admin group may manually override the Status if necessary.

Information about the contract requester, company involved, and locations the contract covers may be added by clicking the Lookup Icon next to those fields. If a desired Requester, Company, or Location does not exist, a record must first be created in order to link it to the Contract.

The **Create and Attach** button at the bottom of the first record tab generates an MS Word copy of the current contract using a template and automatically attaches the file into the **Contract Files** field.



Notes relevant to the contract process are entered into the three notes fields on the Notes and Approvals tab. The **Notes To/From Party Contacts** is used for messages with the contract party. The **Send Notes to Outside Party** button sends the latest note contents to the contract party. The **Approval Notes** field is used for notes pertaining to the approval process. The **Ongoing Notes** field is used for any other notes that apply to the contract.

Information about the renewal process is stored in the Related Contracts/Renewal tab. Fields for capturing the renewal notification date, the renewal cost, the renewal status and renewal notes are provided as a default. If relevant, linked parent and renewal information is automatically updated by the system. Users typically do not link contracts manually.

Configuration items and insurance certificates can be linked to individual contracts through the Insurance/CI's tab.

Insurance Certificates are added by hitting the 'New' button in the related table shown above. When possible, insurance certificates should be created from within the contract record to which they apply. For more information on Insurance Certificates, refer to the section titled *Insurance Certificates Table*.

Depending on the selection in the CI Involvement field, different fields appear for the user to input linked configuration items.



For any number of linked configuration items, use "For One or More CI's". For contracts with no linked configuration items, use "No CI's".

Once the appropriate fields are filled in, the contract may be saved in a Status of Draft to await revision. If the creator knows which approval workflow applies to a particular contract, he or she may also hit the **Submit for Approval** button directly to save the contract and move it into a Status of Pending Approval. For more information on the approval process, refer to *Handling Approvals* below.

## Handling Approvals

The order and nature of approvals are determined according to the Workflow Title selected. Workflow Title choices are determined from the combination of selections in the **Contract Party Type** and **Contract Type** fields. For information on setting up the individual Workflows, refer to the section titled *Workflows Table*.



The Approval Type is automatically set by the system. To submit a contract for approval, select the correct Workflow Title from the drop-down provided and hit the **Submit for Approval** button. This updates the Status of the contract from Draft or Pending Contract Manager to Pending Approval and starts the approval process.



There are two possible Approval Types:

- Parallel
- Sequential

If an approval workflow has an Approval Type of Parallel, the system automatically generates the approvals required according to the information specified in the selected workflow. The approval records for each contract appear in the Notes and Approvals tab under the Approvals Required subsection.

The Approval teams are notified and must edit the pre-generated Approval records to either Accept or Reject the contract as is. Once all required approvals are received, the system automatically sets the contract Status to Approved.

If a workflow has an Approval Type of Sequential, the system automatically notifies the first approvers in the sequence of the pending approval. A progress bar also appears in the Common Area to provide a quick visual reference of the approval process.

To advance in the sequential workflow, an approver may select one of three actions in the Approval Action field:

- Approve and Route Forward to send the contract to the next approver in the sequence
- Reject and Route Back one step to send the contract to the previous approver for changes

- Reject and Route to Beginning if the contract requires significant changes



Once an Approval Action is selected, the **Review Completed** button is hit. The system automatically generates an audit trail of approval records under the Approvals Received subsection. The next approver in the approval sequence is notified that a contract is pending his or her approval.

If a user who is not on the current approval team attempts to approve a contract, the system will throw an error and prevent the user from taking the approval action.

When all the required approvals are received by the system, the contract Status is automatically changed to Approved.

### Handling Related Contracts/Renewals

Related Contracts and Renewals are handled in the Related Contracts/Renewal tab of a contract record.

To create a related contract, select the **New Record Type** of the new contract. Different choices show up in the drop down depending on the **Record Type** of the original contract. These fields are visible only when the contract is no longer in a Status of Draft or Pending Approval. Hitting the **Create Related Contract** button maps relevant information from the current contract into the new one. If the new contract is a renewal, hit the **Create Renewal Contract** button instead. A new contract generated in this way may be edited before saving.



If applicable, the system automatically links Renewal contracts to any preceding contracts, creating a chain for auditability. This information and related information on the contract process for renewals are stored in the bottom half of the Related Records/Renewal tab. Users typically do not enter information into the **Renewal Contract** and **Previous Contract** fields manually.



### Contract Processing

This section covers the Status changes not mentioned in the explanations above.

Once a contract changes to a Status of Approved, the system automatically updates the contract to a Status of Active when the Contract Start Date arrives.

Similarly, when the Contract End Date arrives, if the contract does not have an associated renewal contract, the system automatically sets the Status to Expired. If the contract does have a renewal, the system instead sets the Status to Renewed.

The contract may be canceled by hitting the **Cancel Contract** button in the common area. Additionally, users of the Admin Group have the ability to manually change the contract status to Cancelled.



If the contract has a Status of Approved, two buttons become visible in the Common Area. The **Send Back for Reapproval** button emails the approving teams that reapprovals are necessary, updates the Status to Pending Approval, and restarts the workflow from the beginning. The **Mark as Signed** button updates the contract status to Signed.



### Turning Off Approvals

The Contract Management tables have Approval handling set up by default. The associated processes may be turned off in order to use Agiloft as a contract repository. To turn off Approvals, do the following:

1. Change permissions to allow the Contract Management Group to change the Contract Status field.

2. Remove status-changing buttons from the Layout: Cancel Contract, Submit for Approval, Send Back for Reapproval, Mark as Signed.
3. Remove the Approval Tab and related fields from the Layout.

For help on how to configure Layouts, please refer to the main Agiloft manual.

## Ownership

Records in this table are owned by the Contract Requester. Specifically, a record is owned by the user whose ID matches the number in the Requester ID field. By default, the Contract Requester is the user who created the contract.

## Workflow

The Contracts Table has the following workflow:
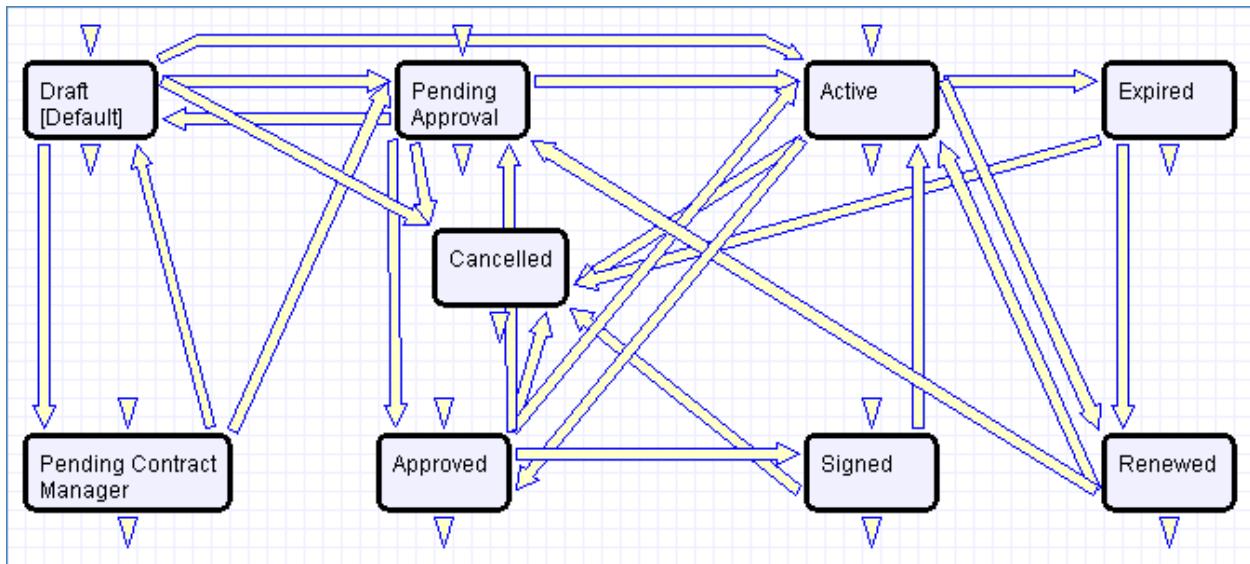


## Reports

The Contracts table has the following Charts and Reports Set up:

## INSURANCE CERTIFICATES TABLE

Purpose: The Insurance Certificates Table is used to hold insurance certificates. Each record in the table represents one insurance certificate that is linked to a contract.

Some time-based rules are disabled by default and need to be enabled for the table to work correctly. To do this, go to the Rules tab of the Setup Contracts page, edit a rule with "(Disabled)" in the title, and click "Yes" in the "Rule is enabled" section at the bottom of the page.

### Use Case

Insurance certificates are created directly from a particular contract or through the Insurance Certificate main table page. Certificates are created with a default Status of Valid. They may also have a Status of Expired or Contract Inactive.

Ten days before the **Expiration Date** arrives, the Contract Management Team is notified of the upcoming expiration.

When the **Expiration Date** arrives, if the associated Contract is still in a Status of Active, the system updates the certificate Status to Expired. An email notification is also sent to the Contract Manager regarding the expiring insurance certificate.

If the parent Contract status is not Active, the system updates the certificate status to Contract Inactive to reflect this.

Each insurance certificate also contains information about the parent contract for reference.

## Ownership

Records in this table are owned by the creator of the certificate. Specifically, a record is owned by the user whose Login matches the login in the Creator Login field.

## WORKFLOWS TABLE

Purpose: The Workflows table holds different pre-determined workflows for the contract approval process.

## Use Case

Workflow records may be created by members of the Admin, Contract Manager, and Change Manager groups.

Required fields are marked by a red asterisk. These include the Workflow Title and the Contract Types. Based on the choice of table in the **Related To** field, there may options further specify to which business processes the workflow applies.

Each workflow may have a Workflow Type of 'Parallel' or 'Sequential'.

If a workflow is Parallel, the required Approvals must be specified. When the workflow is selected for use in an approval process, the system automatically generates required Approval forms using the approval templates specified. New required approval templates are created directly from within the related table by hitting the New button.

If a workflow is Sequential, the steps for the workflow must be specified. Unlike parallel workflows, the system does not generate the entire set of required approval records upfront. Instead, the system generates an Approval record after a decision is made in each step of the sequential process.

New steps can be created directly from within the related table by hitting the New button. For more information on Steps, see the section titled Steps Table.

The records currently using an approval workflow are displayed as a related table in the Related Records tab.

### Workflow Statuses

Each workflow may have one of three Statuses:

- Available
- In Use
- Inactive

A Status of Available indicates a workflow that is not currently in use by any approval process. When in this state, the workflow appears in the drop-down menu for selection. The Administrator may manually change the status to Inactive to if needed.

A Status of "In Use" indicates a workflow that is being used by at least one approval process. The system automatically moves a Workflow into a status of In Use if it is being used. This is done by counting the number of records shown in the Related Records tab. If the number of related records is greater than 0, the workflow is moved into a status of "In Use". Workflows with this Status are not editable in order to ensure business process integrity. If the number of related records is equal to 0, the workflow is moved into a status of Available. For more information on Steps, see the section titled Steps Table.

A Status of Inactive indicates a workflow that is no longer available for use in the approval process. An inactive workflow does not appear in the drop-down menu as a possible approval process selection. The Administrator may change an Available workflow to Inactive if needed.

## Ownership

Records in this table are owned by the user that creates them. Specifically, a record is owned by the user whose Login matches that of the login in the Creator Login field.

Purpose: The Steps table holds the individual steps associated to a particular workflow in the Workflow table. Steps are created only for approval workflows with a **Workflow Type** of 'Sequential'. For more information on Workflows, see the section titled Workflows Table.

## Use Case

Step records may be created by members of the Admin, Contract Management, and Contract Owner groups.

Steps may be created directly from the Steps. However, it is strongly recommended that steps be created from within particular Workflow record via the related table.

Each step must be associated to a particular sequential Workflow in the workflow table. The Workflow must exist prior to creating the step record. Selecting a Workflow title from the dropdown automatically links the step to the correct table: Contracts, Documents, and Change Requests.

To create the step, the user selects the Previous Step Number from the drop-down and enters the current Step Number into the record. The user must also provide a Step Title and an Assigned Team. If applicable, The Next Step Information fields are populated automatically by the system.



The All Steps tab displays all current steps in the workflow as well as the associated progress bar graphic for the individual step. The progress bar is populated automatically by the system.

To minimize error, it is strongly recommended to create Steps in order. In addition, steps should always be created through the associated Steps table from within a particular Workflow record if possible.

Each step has two possible Statuses:

- Deletable
- Non-deletable

A status of Deletable indicates a step in an approval workflow that is not in current use. Specifically, a step is deletable if it is part of a workflow that has been marked as Available or Inactive. A step may be deleted in this state.

A status of Non-deletable indicates a step in an approval workflow that is currently "In Use". Specifically, a step is non-deletable if the workflow it belongs to is in a Status of "In Use". An attempt to delete a non-deletable step is prevented by the system and results in a pop-up explaining the result.

## Workflow

The Steps table contains mostly static data, and thus does not have any associated workflow or actions.

## Ownership

Records in this table are owned by the person that creates them. Specifically, a record is owned by the user whose Login matches that of the login in the Creator Login field.

# DOCUMENTS TABLE

Purpose: This table can be used to manage the creation and publication of documents of various types, from marketing collateral to employee procedure manuals.  A light-weight parallel publication approval process is included.

Examples of documents that may be covered: FAQs, official memos, published company policies, user manuals, newsletters, press releases, and so on.  The table may be used to manage documents that are accessed only through Agiloft (though the records in this table) or documents that are published at the company website, intranet or printed and distributed.  Access to the documents is controlled through permissions based on a choice field within the record.

## USE CASE

### End User Record Submission

An end user belonging to the Document Creator team can create a document through the EUI.  Action buttons will be provided to the end user to move the document through the workflow.

When a user submits a document for review, the contact information fields are automatically populated based on the details in his/her record - including the user's department.  A direct link to the Department and the Department Manager will be auto-populated based on the submitter's department.

The record will be created in a default status of Draft.  After supplying the required information and uploading a document, the user will click the Submit for Review button to begin the review process.  The status of the record will be updated to Pending Review.  If the user is not prepared to submit the record immediately, then can save the record and make further updates.

### Technician Record Submission

Staff users in the Document Management, Admin and Document Creator groups can submit documents.  Only Admins and Document Managers can update the status of a Document record manually.  All other groups will use Action Buttons to move the document through the workflow.

The record will be created in a default status of Draft.  After supplying the required information and uploading a document, the user will click the Submit for Review button to begin the review process.  If the user is not prepared to submit the record immediately, they can save the record and make further updates.

### Processing of Records

After a document record is submitted for review, the Assigned Team (Document Management Team) is notified and assigned the record by default.  A Document Manager will review the document for content, formatting and to determine if the document requires additional review.

If there are any issues with the initial document, the Document Manager will click the "Return to Submitter" action button to send the document back to a status of Draft.  A rule will run to notify the submitter that the document requires revision prior to submitting it for approval.  The user will make the appropriate updates and submit the document for review once again.

If there are no issues and the document does not require additional reviews, the Document Manager will click the "Publish" action button. This will initiate a notification to the submitter that the document has been published.

If the document requires additional reviewers, the Document Manager will select the appropriate reviewers by adding reviewer names to the Potential Reviewers field. This field is a link to a single field (Full Name) with multiple values enabled in the Employee table, displayed as multiple value box with a popup selection list and filtered to individuals belonging to the Document Reviewers team. A search view will be provided to allow the Document Manager the ability to quickly see and sort on specific fields such as Department.

After selecting the appropriate reviewers, the Document Manager will click a button to Submit for Approval or change the status of the record to Pending Approval.

When the record is saved, an approval record is created for each of the reviewers through a conversion process from the reviewer's Employee record. A linked record action will update the Last Document ID field in the Employee record. The system will sense a change in this field and start the conversion of the Employee record to an Approval record. The Approval record will be linked to the Document record through the mapping of Last Document ID field to the linked Document set in the Approval record. Additional actions will prevent the creation of multiple approval records allowing additional reviewers to be added during the approval process.

When an approval record is created, the reviewer is automatically notified that they have a document to review and approve. The linked set from the Document record in the Approval record will include a hyperlink to the document using the "view only source field display" for the attachment field. The reviewer can click the link to launch the document – allowing the reviewer to mark up the document which can then be uploaded to the approval record (reviewers will not be able to upload the document to the source record). All updates to the document will be done by the submitter, the document manager or an admin.

The approval record is created with a default Approval Status of Pending Approval. The reviewer will either approve or reject the document and provide approval notes. If the approval record is rejected, the reviewer must provide approval notes explaining why the document is rejected.

When all of the approval records have been approved or rejected, based on a calculation of the related Approvals in the Document record, the review process is essentially completed. If any rejections were received as part of the review process, the document record will be returned to the submitter with a status of Draft and the approval records will be updated to a status of Requires Reapproval.

The submitter can review the comments made by the reviewers and see any the red-lined documents that have been attached to the approval records. Once the changes have been made the submitter can begin the review process again by once again clicking the Submit for Review button.

Once the Document Manager has reviewed the changes, he can begin the approval process again by changing the status to Pending Approval or clicking the button. This will set the existing approval records to Pending Approval with a notification to the reviewers.

*Example of Document Record Details tab.*

If there are no rejections, the status of the Document record is updated to Ready for Publication with a notification to the Submitter and the Document Management team.

The Document Manager can then attach the final approved document to the Published Files field and update the status of the Document to Published.   A notification will be sent to the submitter.
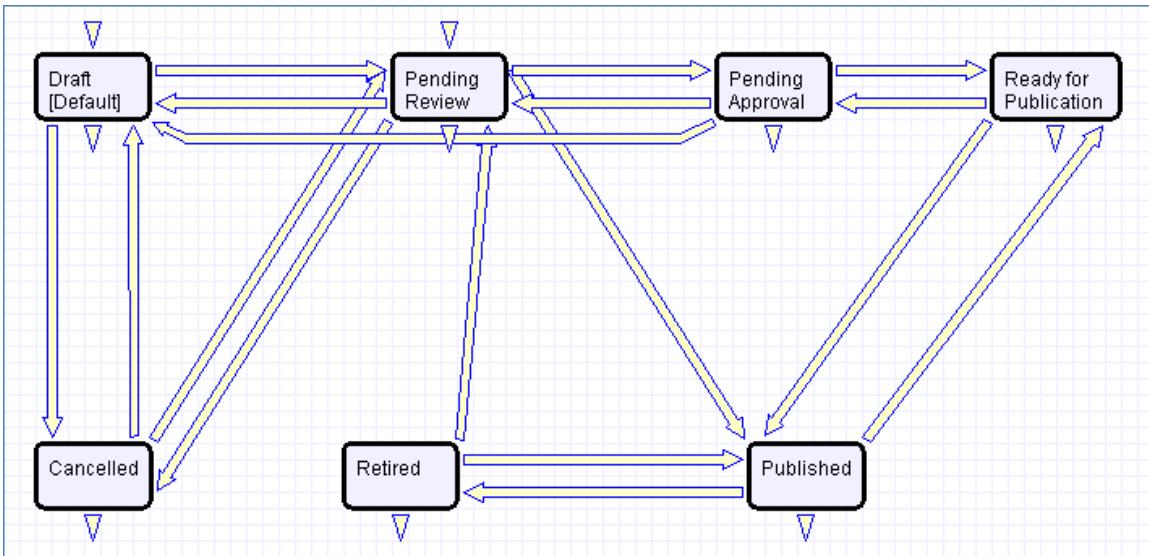
*Workflow for Documents table.*

## PROJECT MANAGEMENT TABLES

### PROJECTS TABLE

Purpose: This table holds records for project management activities.  It is currently optimized for companies providing consulting services to their clients, and allows them to manage their billable and unbillable hours, work authorizations, and overall project status.  It can easily be customized for internal project management instead.

#### Use Case

Members of the Admin or Project Manager groups may manually create Project records.  Projects are creatable only in the "Planned", "Assigned", "Work in Progress" and "Awaiting Customer Feedback" states.

Only members of the Product Manager and admin Groups may edit others' Project records, but Support Staff (Base ServiceDesk group) may view their own Project records. The user who created the Project is automatically set as the Project Manager, and will get email notifications pertaining to the Project's status, such as when all tasks are completed or the hours spent on the project have exceeded what was authorized. Project CCs can be specified who will also receive notice when the Project is completed. The information on the "Contact Information" tab will be filled out with the information of the Project Manager's Manager, provided by the Employee record of the Project Manager.

Project records are divided into two broad categories: Internal and Client-Related. These categories are further divided by type. The tasks that are automatically generated by a project depend on the type selected. For each Project type there are task templates that list the tasks that get automatically generated. The exact selection of tasks that will be generated can be specified on the "Tasks" tab of the Project record. Ad-hoc tasks can also be created using an action button on that tab. Additional fields are

visible when a Project's Category is "Client-Related". These fields hold information about customer contracts, contact details, and authorized hours.



*Main tab of Project record.*

Time spent on the project is tracked on the "Time / Billing" tab. It includes a small form to enter time spent and a related table of time entries for this project. Time entries can be searched for in the related table and linked to the project manually.

## Workflow
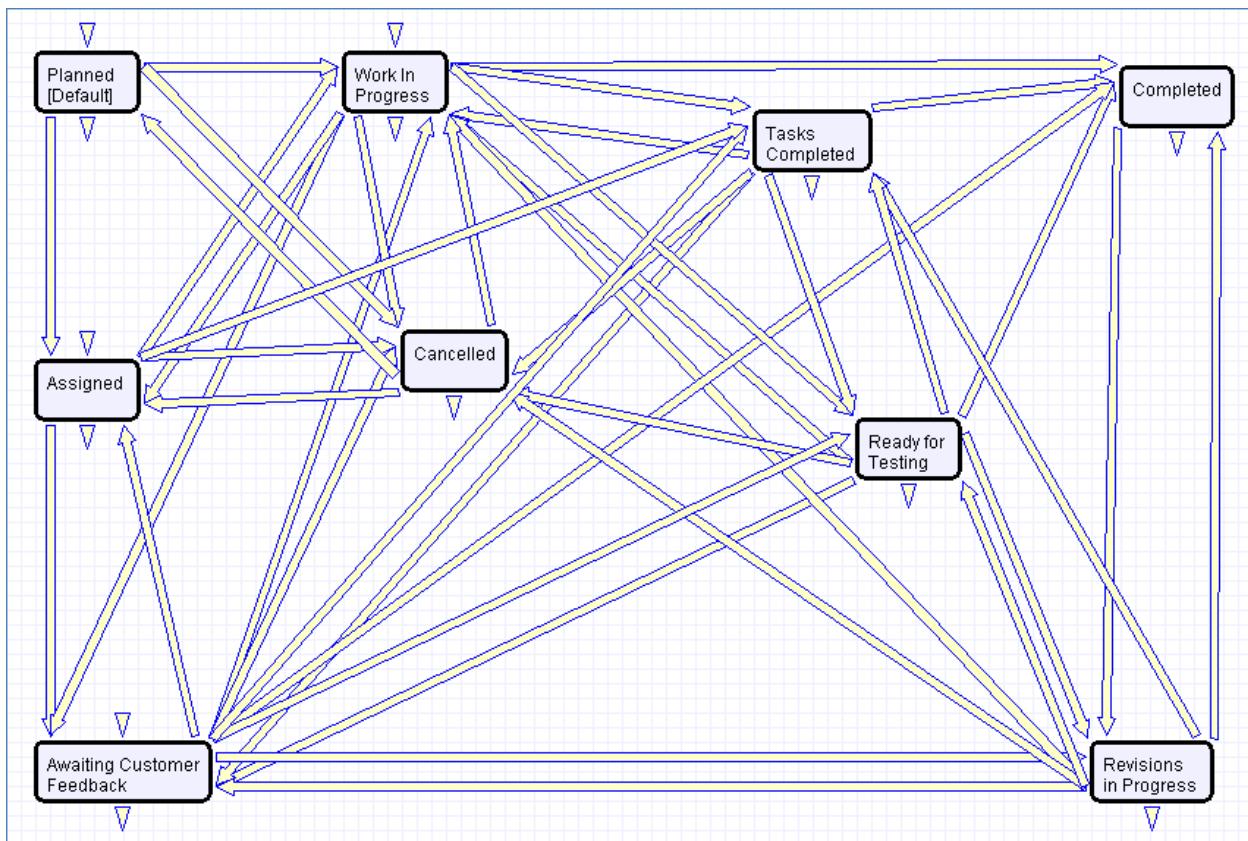


*Default workflow for the Projects table*

## Ownership

Records in this table are "owned" by the Employee designated as the Project Manager.

## PROJECT TYPES TABLE

Purpose: This is a background table that holds the Project Types referenced by the Projects table. It allows the creation of new Project Types by Project Managers and other users without Admin group privileges. If there are any Tasks that are required for a given Project Type they can be linked to the Project Type in the Task Templates table.

## PURCHASE ORDERS (PO) TABLE

Purpose: The Purchase Order table tracks authorized billable hours for a project.  It is shown as a related table within the Project table.  It could easily be linked to support cases or Quotes or other tables within the system.

### Use Case

PO records are creatable manually via the web form, from within Project records in the related table, or via mass import.  Only members of the Professional Services, Sales and admin Groups may create or import records. PO records are creatable in any workflow state.

Only members of the Professional Services, Sales and admin Groups may edit records, but Support Staff may view all PO records.
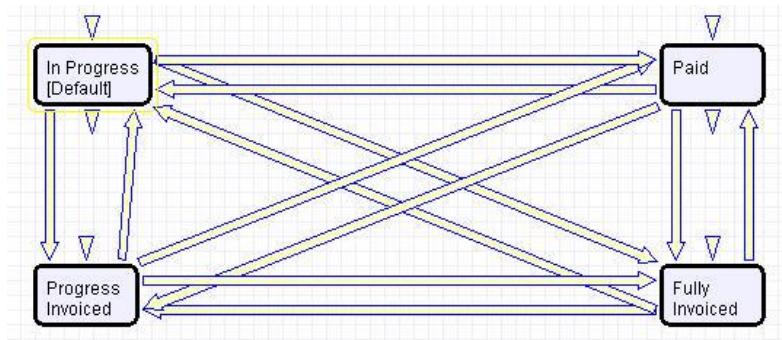


*Default workflow for the Purchase Orders table*

### Ownership

Records in this table are "owned" by an Employee, generally the one who created the record.  Each record is associated with a particular Employee login.

## SUPPORT CASES TABLE

Purpose: This table is used to manage external customer support requests.  It is set up to be accessed by the Customer group and the Base Technical Group.

### USE CASE

## End User Record Submission

Customers may create support cases using the tab in the end user interface or by sending an email (once an inbound email address is set up).

When a customer submits a support case, the contact information fields automatically populate based on the details in his/her user record.  If the user record doesn't contain a value in the Customer Name or Email fields, the customer will be required to enter a value in those fields manually.

The Type of Issue is set by default to Question.  If the customer changes it to Installation Issue or Bug, they will be required to fill out the Steps to Reproduce field.  The case is assigned by default to the Support Team and the default status is Open.

## Technician Record Submission

Staff members may also submit support cases on behalf of a customer, associating the customer with the case.  If an internal staff user creates a case, he may assign it directly to an individual or a team other than the Support Team and may set its starting status to Open, Assigned, or Closed.

When the record is created, emails are sent to the customer acknowledging receipt of the support case and to the assigned team (or person) telling them the case has been assigned to them.

If a Support Staff technician creates a record in a status of Closed an email is sent to the customer telling them how to reopen their case.

Workflow actions send these emails automatically, but staff users can override them if given permission to override workflow actions.

## Processing of Records

When a technician works on a case, if he needs more information from the customer in order to take further action, he can set the status to Sent to Customer.  This will automatically send an email to the customer requesting further information and include the content of the Additional Notes field, an append-only field that is used to communicate with the customer.  The email includes a hyperlink for the customer to click to login to edit the case directly.

When the customer edits the case or replies to the email, the status changes to Updated by Customer and an email notifies the assigned person that the customer has replied.  The customer is able to update the Additional Notes field directly and any text from an email reply to a system email maps to that same field.

If the customer updates the case at any point, an email notifies the assigned person of the update.

If the technician needs to reassign the case to someone else, he or she simply changes the Assigned Person field to that person's name and the system will email the new assignee notifying them of the reassignment.

The Staff Only Notes field holds working notes that should not be visible to the customer.

When the technician has completed work on the case, he/she sets the Status field to Closed and puts the solution notes into the Solution field.  This triggers an email to the customer that includes the content of the Solution field and tells the customer that the work is done.  This closing email gives the customer a hotlink back to the record if they wish to reopen it and instructs them to explain why they are not satisfied with the solution.  Clicking the hotlink will automatically change the "I Would Like To

Reopen My Ticket" field to Yes, which in turn sets the Status of the ticket to Reopened and notifies the assigned person.

By default no escalation rules are set up for the Support Case table.
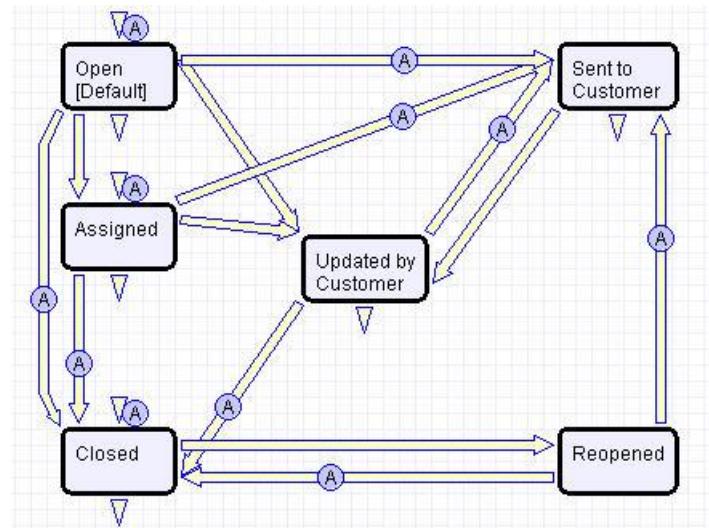


*Support Case table default Workflow*

## OWNERSHIP

Records in this table are "owned" by the individual customer. This means each record is associated with a particular customer login and no other customers will be able to edit that record, though members of the Customer Manager group can view all records submitted by other people at their company.

# SALES/CRM TABLES: LEADS TABLE

Purpose: This table is used as the initial point of entry for sales leads. Leads may self-register at your website or may be imported from a lead generation program. The initial qualification is done in this table, which contains all the fields desirable for managing sales.

Once a lead is qualified, it may either be fully worked in this table, or converted into a contact, an opportunity, and a company/account record, and the sales process may be managed in those records.

## USE CASE

### Record Creation

Leads can be created directly using the web form, or, once an inbound email address is set up, via email. New leads may only start out as Qualified or Unqualified in the workflow. The Leads table is set up by default to allow those in the Guest group to create records, allowing "click to register" lead generation hotlinks, and the embedding of the lead creation form in a web page.

Naturally leads may also be imported from a spreadsheet from a lead generation program.

## Processing of Records

Admins and members of the Sales group can create, view, and edit Leads.  No other groups have access to the table by default.

When a lead's status changes to "Converted", Agiloft converts the information in the record into three new records in the three other tables: Company, Opportunity, and Contact. This order of creation is important because both the Opportunity and Contact records will contain links to the original Company via the new record.  If the Company record is not created first, the Contact and Opportunity records will be unlinked and orphaned, disabling reporting features.

Data fields containing information relevant to the company, such as address and billing address, company website, industry, annual revenue and number of employees, map to the Company record. Sales-specific data fields, such as key requirements, earliest and latest possible close date, and sales actions taken, map to the Opportunity record.  All data from the lead referencing a specific person at the company map into a new Contact containing the individual's desk and cell phone numbers, email address, email preferences, work hours, and so on.
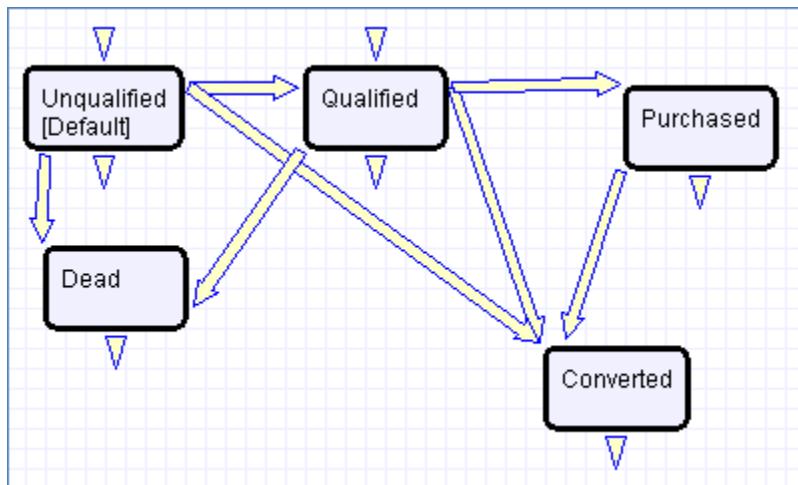


*Default workflow for Leads table*

## OWNERSHIP

Records in this table are "owned" by the individual assigned sales rep, so each record is associated with a particular user login.  Only members of the Sales and admin groups can view or edit Leads.

## SALES/CRM TABLES: OPPORTUNITIES TABLE, DEALS SUB-TABLE

Purpose: This table tracks sales opportunity information and contract information when an opportunity becomes a sale.  An opportunity, when closed, is simply changed into a Deal, and all information is maintained, while new contract fields appear.

## USE CASE

Most Opportunity records are created from Leads via a Conversion rule.  Deals are typically created from Opportunities when they change to "Closed Won".  Members of the Admin or Sales groups may create records manually.

An Opportunity record should be considered a possible sale, while a Deal is an actual finalized and confirmed agreement.

Admins and members of the Sales group can create, view and edit Opportunities.  Members of the Professional Services and Support Staff groups may view all Opportunities. No other groups have access to the table by default.

The Opportunities table tracks information on possible sales for your organization. The Deals section handles Opportunities that have been successfully closed and turned into contracts and contains specialized fields for each contract's unique information, such as SLA requirements, renewal dates, and so on.

When an Opportunity's status changes to "Closed Won", a rule converts it to a new Deal record.
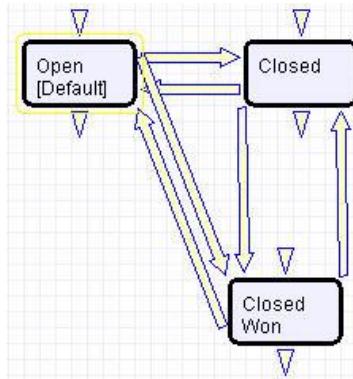


*Default workflow for the Opportunities Table*
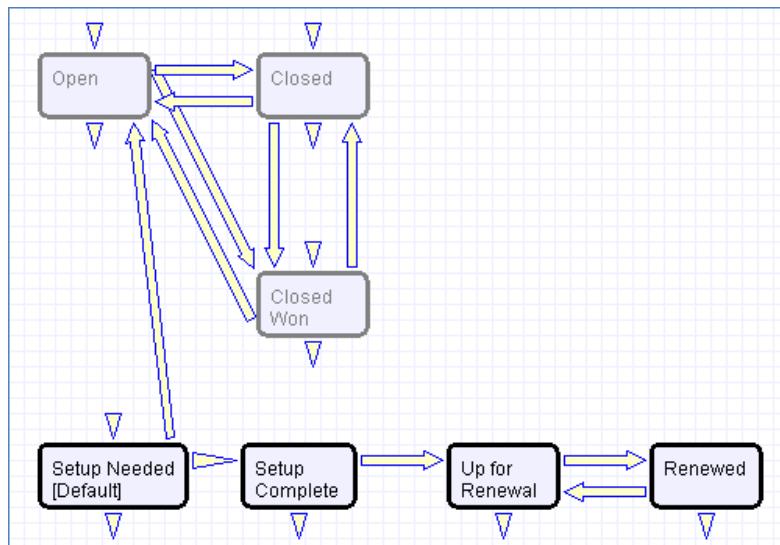


*Default workflow for the Deals sub-table*

OWNERSHIP

Records in this table are "owned" by the individual assigned sales rep, so each record is associated with a particular user login from the Contacts table.

# APPENDIX A – DETAILED TABLE SETUP

In this Appendix we provide detailed technical information on the fields in each table, the rules and actions that are preconfigured and other details of the out-of-the-box implementation.

The tables showing all fields in each table are intended to explain the purpose of some system default fields and to give a quick reference for how they might be used.

The rules and actions that run in the background to validate that users follow the correct processes and to automate process steps and notifications are documented with screenshots and details.

This automation is described in more general terms in the Use Case section of the main document for each table.

The tables are listed alphabetically here.

## APPROVALS TABLE AND APPROVAL TEMPLATES SUBTABLE

### APPROVAL AND APPROVAL TEMPLATES FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|
| Approval Notes | Text | Maximum: 5000, 95 x 5 display. |
| Approval Running Notes | Append Only Text | This field will be used to hold Approval Notes for parallel approvals. When Approval notes are added to an approval record, the notes are copied into this field and into the contract approval notes field as well. Visible for only those records whose source is Approval Template. |
| Approval Status | Choice | Choice: Pending Approval, Approved, Rejected, Requires Reapproval. Default: Pending Approval |
| Approval Team | Link to selected fields from other table:Teams Allow multiple values:No Allow entries not in source table:No | |
| Approved/Rejected By, Approver Primary Team, Approver Teams | Link to selected fields from other table:People Allow multiple values:No Allow entries not | Person who changes the approval status to approved or rejected. |

| | in source table:No | |
|---|---|---|
| Approver | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | Filtered to person on the Approval Team |
| Contract Approval Action | Choice | Mapped via conversion for Source = Sequential Approvals, visible only if Source is Sequential. Used by rule to set the Approval Status |
| Contract Assigned Team, Contract Company Name, Contract Amount, Contract Description, Contract Files, Contract Start Date, Contract Title, Contract ID, Contract Owner, Contract Requester Email, Contract Requester Name, Contract Workflow | Link to selected fields from other table:Contracts Allow multiple values:No Allow entries not in source table:No | |
| Created By, Creator Login, Creator Team | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | |
| Date Approved/Rejected | Date/Time | Date the Approval Status field was set to Approved or Rejected |
| Document ID Document Status Document Title Document Type Document Subtype Document Priority Document Description Document Purpose Document Access Document Audience Document Source Files | Link to selected fields from other table: Documents Allow multiple values: No Allow entries not in source table: No | Linked set for Document requiring approval.  Visible only if Related to is Documents.  Use the "view only source field display" for the attachment field to provide hyperlink for reviewer. |
| Documents | File with versioning | Used by the approver to supply marked up documents as part of the approval process.  Versioning enabled; check in/out fully enforced; visible if Related To = Documents |

| Related To | Choice | Update existing 'workflow related to' choice list with Documents |
|---|---|---|
| Source | Choice | choice: Sequential Workflow, Approval Template - Default: Sequential Workflow |
| Source for Template Conversion | Choice | Used in the conversion action to populate the Approval value. Don't delete. Default value: Approval Template |

## APPROVAL RULES AND ACTIONS

The Approvals table has the following rules set up.

### Create: Contract Approval Creation Actions for Both Workflow Types

**Purpose:** This rule handles actions resulting from the creation of an Approval Record generated from a document, sequential workflow or parallel workflow (i.e. approval template). It does not control the creation of any records itself.

**Runs:** Upon creation by Web or API

**Search Filter:** R: Related to Contracts or Documents

**Priority:** 40

**Actions:**

#### I: All Create Actions

```
if (Source='Sequential Workflow') {
        if (Contract Approval Action='Approved - Route Forward') {
                Set Status to Approved and people fields
        } else {
                Set Approval Status to Rejected and people
        }
} else if (Source for Template Conversion='Approval Template') {
        Email Approval Team of Pending Approval
} else if (Source='Document Management') {
        Set Approval Team to Doc Reviewers
        Email Reviewer of Pending Approval
}
```

### Edit: All Edit Validations

**Purpose:** This rule handles validations for approval records associated to parallel workflows. It checks against the teams of the user editing the record.

**Runs:** Upon edits by Web

**Search Filter:** None

**Priority:** 10

**Actions:**

### I: All Edit Validations

```
if (Approval Status changed from: 'Pending Approval' to: any of 'Rejected','Approved' last user's modification and Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team' and
(Source for Template Conversion='Approval Template' or Source='Document Management')) {
    You are not on the approval team
} else if (Approval Status changed from: any value to: 'Rejected' last user's modification and Approval Notes not changed
last user's modification) {
    Need to add notes when rejecting
}
```

## Edit: All Edit Action for Updates by Web or API

**Purpose:** This rule handles all updates to an approval record and notifications resulting from changes in the parent contract record.

**Runs:** Upon edits by Web or API

**Search Filter:** None

**Priority:** 30

**Actions:**

### I: All Edit Actions

```
if (Related To='Contracts') {
    if (Source for Template Conversion='Approval Template') {
        if (Approval Notes changed from: any value to: any value during record's last modification) {
                Copy approval notes to contract
                Copy approval notes to running notes
                Blank approval notes
        }
        if (Approval Status changed from: any of 'Approved','Pending Approval' to: 'Rejected' during record's last
        modification) {
                Set Contract Status to Draft
                Set other Approvals to Requires Reapproval
                Email Contract Manager of rejection
        } else if (Approval
        Status changed
        from: any of
        'Rejected','Requires
        Reapproval' to:
        'Pending Approval'
        during record's last
        modification) {
                Email Approval Team of Pending Approval
        }
        if (Approval Status changed from: any value to: 'Approved' last user's modification) {
                if (Approver=NULL) {
                        Update Approver field if blank
                }
        }
    }
}
if (Related To='Documents') {
    if (Source='Document Management') {
        if (Approval Notes changed from: any value to: any value during record's last modification) {
                Copy approval notes to Document
                Copy approval notes to running notes
                Blank approval notes
        }
        if (Approval Status changed from: any of 'Rejected','Requires Reapproval' to: 'Requires Reapproval' last user's
        modification) {
                Email Reviewer of Pending Approval
                Blank approval notes
        } else if (Approval
        Status changed
        from: any value to:
        'Requires
        Reapproval' last
        user's modification)
        {
                Email Approver Requires Reapproval
        }
    }
}
```

## Edit: By API Retain Rejection

**Purpose:** This rule ensures rejection records are properly handled when a contract cycles through an
approval process.

**Runs:** Upon edits by Web

**Search Filter:** R: Status changed from Rejected to Requires Reapproval

**Priority:** 20

**Actions:**

> ### U: Set Status to Rejected
>
> **Updated Field:** Approval Status
>
> **Updated Value:** Rejected

## APPROVAL TEMPLATES RULES AND ACTIONS

The Approval Templates subtable has the following rules set up.

### Edit: Approval Record Creation

**Purpose:** This rule handles the creation of an approval record from an approval template when the relevant parallel approval workflow is selected.

**Runs:** Upon edits by Web or API

**Search Filter:** LF: Last Contract ID just changed

**Priority:** 1

**Actions:**

> ### C: Create Approval Record
>
> **Convert to:**  Approval Table
>
> **Options:** Silently with no confirmation; do not report errors in GUI

## CALENDARS TABLE

### FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments |
|---|---|---|
| *Description | Short Text | A short title or description |
| *Duration | Elapsed Time | *Defaults to 0 days 0 hours 30 minutes*<br><br>For determining the length of the calendar event |
| *Start Time | Date/Time | *Defaults to the record creation date* |

| | | For listing the event start time |
|---|---|---|
| Linked Records | Short Text | When a Calendar record is created via the calendar icon from somewhere else, creates a clickable one-way link to the original record. |
| Notification Time | Elapsed Time | *Defaults to 0 days 0 hours 15 minutes*<br><br>Determine when to send an email reminder |
| Personal Invitations | Link to single field from User table: Full Name | For inviting individuals; Note that more than one value can be selected |
| Team Invitations | Link to single field from Team table: Team Name | For inviting whole teams. Note that more than one value can be selected |

## CHANGE REQUESTS TABLE

### FIELDS

| Field Label | Data Type | Comments |
|---|---|---|
| 1$^{st}$ Approver | Link to User: Full Name | Used to identify the first approver |
| 1$^{st}$ Date Approved/Rejected | Date/time | Default: Date when the Approved by 1$^{st}$ Approver field changed |
| 1$^{st}$ Approval Team, 1$^{st}$ Team Manager | Link to Team: Team Name, Team Leader | Used to select the team that should approve |
| 2$^{nd}$ Approver | Link to User: Full Name | Used to identify the second approver, visible only if Number of Approvals needed is > 1 |
| 2$^{nd}$ Date Approved/Rejected | Date/time | Default: Date when the Approved by 2nd Approver field changed, visible only if Number of Approvals needed is > 1 |
| 2$^{nd}$ Approval Team, 2$^{nd}$ Team Manager | Link to Team: Team Name, Team Leader | Used to select the team that should approve, visible only if Number of Approvals needed is > 1 |
| 3$^{rd}$ Approver | Link to User: Full Name | Used to identify the third approver, visible only if Number of Approvals needed is > 2 |
| 3$^{rd}$ Date Approved/Rejected | Date/time | Default: Date when the Approved by 3rd Approver field changed, visible only if Number of Approvals needed is > 2 |
| 3$^{rd}$ Approval Team, 3$^{rd}$ Team Manager | Link to Team: Team Name, Team Leader | Used to select the team that should approve, visible only if Number of Approvals needed is > 2 |
| 4$^{th}$ Approver | Link to User: Full Name | Used to identify the fourth approver, visible only if Number of Approvals needed |

| | | is > 3 |
|---|---|---|
| 4<sup>th</sup> Date Approved/Rejected | Date/time | Default: Date when the Approved by 4<sup>th</sup> Approver field changed, visible only if Number of Approvals needed is > 3 |
| 4<sup>th</sup> Approval Team, 4<sup>th</sup> Team Manager | Link to Team: Team Name, Team Leader | Used to select the team that should approve, visible only if Number of Approvals needed is > 3 |
| 5<sup>th</sup> Approver | Link to User: Full Name | Used to identify the fifth approver, visible only if Number of Approvals needed is > 4 |
| 5<sup>th</sup> Date Approved/Rejected | Date/time | Default: Date when the Approved by 5th Approver field changed, visible only if Number of Approvals needed is > 4 |
| 5<sup>th</sup> Approval Team, 5<sup>th</sup> Team Manager | Link to Team: Team Name, Team Leader | Used to select the team that should approve, visible only if Number of Approvals needed is > 4 |
| Actual End Time | Date/time | Used to indicate when the work was actually completed |
| Actual Start Time | Date/time | Used to indicate when the work was actually started |
| Add Time | Action Button | Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the current Change Request record |
| All Time Entries | Related table: Time Entries | Shows all time entries related to this request |
| All Time Spent | Calc on related table | Totals the hours spent on this request |
| Approval Notes | Append only text | For making notes about the approval process |
| Approved By | Link to Employee table: Full Name | Default: Person who changes the Approved by 1<sup>st</sup> Approver field to Yes |
| Approved by 1<sup>st</sup> Approver | Choice: yes/no | This is the field that the first approver changes to approve or reject the request |
| Approved by 2<sup>nd</sup> Approver | Choice: yes/no | Second approver uses this field, visible only if Number of Approvals needed is > 1 |
| Approved by 3<sup>rd</sup> Approver | Choice: yes/no | Third approver uses this field, visible only if Number of Approvals needed is > 2 |
| Approved by 4<sup>th</sup> Approver | Choice: yes/no | Fourth approver uses this field, visible only if Number of Approvals needed is > 3 |
| Approved by 5<sup>th</sup> Approver | Choice: yes/no | Fifth approver uses this field, visible only if Number of Approvals needed is > 4 |
| Assigned Person | Link to Single field in Employee: Full Name | |
| Assigned Team Assigned Team Leader | Link to selected fields in Team: Team Name, Team Leader | |
| Attached Files | File | |
| Backout Plan | Text 1000/90x3 | Required field |
| Business Justification | Text 5000/90x3 | |

| Change Category | Choice:<br>Standard (0)<br>Minor (1)<br>Significant (2)<br>Major (3) | Required |
|---|---|---|
| Change Description | Text  1000 90x6 | |
| Change Summary | Short Text 90/90 | Required, Summary field |
| Child CRs | Related table: Change Requests | Shows all CRs whose parent CR is this one |
| CI Class, CI Name, CI Tag, CI Type, CI ID, CI Responsible Team, CI Serial Number, CI User | Link to Configuration Item table | Displayed if Related to = Configuration Item |
| CI's for Child CRs | link to CI Name with multiple values enabled | Used to select CI's for which Child CR's should be generated and linked back to a parent CI.  Visible only if Related to= Parent of change to multiple CIs |
| Closed By | Link to Employee: Full Name | Default: Person who changes status to Closed |
| Communications | Default Communications table | |
| Conv ID | | System field |
| Conversion child mapping | Choice | Used by conversion, do not delete |
| CR Type | Choice: Independent, Parent, Child | Default: Independent, choices are dependent on Related to values |
| Created By<br>Creator Login<br>Creator Team | Default link to User | |
| Date Closed | Date Time | Default: Date when status changed to Closed. |
| Date Created | Default Date/Time | |
| Date Updated | Default Date/Time | |
| Deletable | Default Choice: Yes/No | Default: No |
| Description of Service<br>Give Responsible Team Priority over CI Team<br>Service Responsible Team<br>Service Title<br>Service Categories<br>SLA Hours - Critical Priority<br>SLA Hours - Standard Priorities<br>Special Instructions | Link to Services: Description of Service<br>Give Responsible Team Priority over CI Team<br>Service Responsible Team<br>Service Title<br>Service Categories<br>SLA Hours - Critical Priority<br>SLA Hours - Standard Priorities<br>Special Instructions | Required Fields, Pulls in values for SLA and responsible team from Service |
| Estimated Time to Complete in Hours | Elapsed Time | |

| History | | |
|---|---|---|
| ID | | |
| Linked Problems | Related table: Problems | Shows all problems linked to this CR |
| Linked Project | Link to Projects table | Used to link a CR to a particular project, visible only if Related to=Project |
| Number of Approvals Needed | Link to Service: Number of Approvals needed | This is a loose linked field and populates with the default value for the service, but users with privileges may be permitted to change it based on circumstance. It defines how many of the approval fields are shown on the layout. |
| Number of Assignees | Floating point | Auto-populated by rule |
| Number of Reopens | Floating point | Auto-populated by rule |
| Number of Teams Assigned | Floating Point | Auto-populated by rule |
| Parent CR Approved, Parent CR Approved By, Parent CR Assigned Team, Parent CR Summary, Parent CR ID, Parent CR Requester, Parent CR Service Title, Parent CR Status | Link to CR: Parent CR Approved, Parent CR Approved By, Parent CR Assigned Team, Parent CR Summary, Parent CR ID, Parent CR Requester, Parent CR Service Title, Parent CR Status | Required Fields, Visible only if CR Type is Child, shows information about the parent CR |
| Related Incidents | Embedded Search Result | Shows all incidents linked to this CR |
| Related to | Choice: Configuration Item, Parent of change to multiple CI's, Project, Other | Required Field, Controls visibility of CI and other fields |
| Replaced CI | Link to selected fields in CI: CI Name | Only show if Service is CI Replacement |
| Requested Date of Completion | Date Time | |
| Requestor Department, Requestor Phone, Requestor Email, Requestor Name, Requestor Manager Email, Requestor Manager | Link to User table | Required Fields, Default: CR Creator |
| Risk if Not Done | Choice: Low, Medium, High, Very High | Required |
| Scheduled End Time | Date/time | |
| Scheduled Start Time | Date/time | |
| Scheduled Time to Complete | Elapsed time | Measures distance between Scheduled start and end times |
| Service Category | Choice: shows all service categories for CRs | |

| | | |
|---|---|---|
| Service Requests | Related table: Service Requests | Shows all Service requests linked to this CR |
| Set CI to Active | Action button | Updates the linked CI to change its status to Active |
| Set CI to In Maintenance | Action button | Updates the linked CI to change its Status to In Maintenance |
| Spawn Child CR | Action button | Converts the CR to multiple child CRs linked to the CI's selected in the CIs for Child CRs field |
| Spawning Problem ID | Link to selected fields in Problem: ID | For conversion linking, will hold the ID of the Problem record that was converted to this Change Request |
| Spawning SC ID | Link to fields from Support Case | Holds ID of Support Case from which this CR was created |
| Spawning SR ID | Link to selected fields in Service Request: ID | For conversion linking, will hold the ID of the Service Request record that was converted to this Change Request |
| Status | wfstate | See workflow diagram |
| Support Cases | Related table: Support Cases | Shows any linked Support Cases |
| Time between start and finish | Elapsed time | Measures time between Actual Start and Actual End Times |
| Time Description | Text | Hold information for time entry |
| Time Spent | Elapsed time | Allows user to enter time spent on this CR |
| Time to Start | Elapsed time | Time from Date Created to Actual Start Time |
| Total Time to Close | Elapsed Time | Measures time from Date Created to Date Closed |
| Type | Default table type | |
| Updated By Updater Login Updater Team | | |
| Urgency of Change | Choice: Standard, Emergency | |
| Working Notes | Append only text | Holds all working notes |

## SAVED SEARCHES

My Assigned:

      Assigned Person = global.name OR Assigned Team contains global.my_teams

Forward Schedule of Change:

      Status != Closed, Sort By Requested Date of Completion, ascending

      Summary: Sum of Estimated Time to Complete

Accomplished Change:

      Status = Closed, Sort By Date Closed, ascending

      Summary: Average of the Total Time to Close

My Approval Needed:

> Any of 5 Approvers = $global.my_full_name and that approval = $null

# RULES AND ACTIONS

## Rule: Change Request – All Creation actions

Runs when created by any method, no saved search

### I: Validations and Conditional Actions

```
if (Spawning SR ID!=NULL) {
      Update "Additional Notes" in Spawning SR
}
if (Spawning Problem ID!=NULL) {
      Update "Additional Notes" in Spawning Problem
}
if (Parent CR ID!=NULL) {
      Set CR Type to Child
}
if (Related to='Configuration Item' and CI Name=NULL) {
      Must select a CI
}
```

### I: Assign Change Request

```
if (Give Responsible Team Priority over CI Team!='Yes' and Assigned Team=NULL and CI Responsible Team!=NULL) {
      Set Assigned Team to CI Responsible Team
} else if (Assigned Team=NULL and Service Responsible Team!=NULL) {
      Set Assigned Team to Service Responsible Team
} else {
      Set Assigned Team to 1st Level Support Team
}
```

### I: Send Emails to Requestor and Assignee

```
if (Status!='Closed') {
      Send Acknowledgment Email to Customer
      if (Assigned Person!=NULL) {
            Send New CR email to Assigned Person
      } else {
            Send New CR email to Assigned Team
      }
}
```

## Rule: CR – Approvals

Runs when created or edited, with API, Email, or Web, no search

### I: Approver Validations

```
if (Approved by 1st Approver changed from: any value to: any of 'No','Yes' last user's modification and 1st Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 1st Approver
}
if (Approved by 2nd Approver changed from: any value to: any of 'No','Yes' last user's modification and 2nd Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 2nd Approver
}
if (Approved by 3rd Approver changed from: any value to: any of 'No','Yes' last user's modification and 3rd Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 3rd Approver
}
if (Approved by 4th Approver changed from: any value to: any of 'No','Yes' last user's modification and 4th Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 4th Approver
}
if (Approved by 5th Approver changed from: any value to: any of 'No','Yes' last user's modification and 5th Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 5th Approver
}
```

Note: Matches against $GLOBAL.my_teams variable.

### I: Status Change Validations

```
if (Status changed from: any value to: 'Approved' last user's modification) {
     if (Approved by 1st Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=2 and Approved by 2nd Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=3 and Approved by 3rd Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=4 and Approved by 4th Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=5 and Approved by 5th Approver!='Yes') {
          Need all approvals
     }
}
```

### I: Approval Notification Actions

```
if (Status changed from: any value to: 'Pending Approval' last user's modification) {
    if (1st Approver!=NULL) {
        Send 1st Approver email
    } else if (1st Approval Team!=NULL) {
        Send 1st Approval Team email
    }
    if (2nd Approver!=NULL) {
        Email 2nd approver need approval
    } else if (2nd Approval Team!=NULL) {
        Email second approval team need approval
    }
    if (3rd Approver!=NULL) {
        Email 3rd approver need approval
    } else if (3rd Approval Team!=NULL) {
        Email 3rd approval team need approval
    }
    if (4th Approver!=NULL) {
        Email 4th approver need approval
    } else if (4th Approval Team!=NULL) {
        Email 4th approval team need approval
    }
    if (5th Approver!=NULL) {
        Email 5th approver need approval
    } else if (5th Approval Team!=NULL) {
        Email 5th approval team need approval
    }
}
if (Status changed from: any value to: 'Approved' during record's last modification and Updated By!='$assigned_person') {
    if (Assigned Person!=NULL) {
        Send CR Approved email to Assigned Person
    } else if (Assigned Team!=NULL) {
        Send CR Approved email to Assigned Team
    }
} else if (Status changed from: any value to: 'Rejected' last user's modification and Updated By!='$assigned_person') {
    if (Assigned Person!=NULL) {
        Send CR Rejected email to Assigned Person
    } else if (Assigned Team!=NULL) {
        Send CR Rejected email to Assigned Team
    }
}
```

## Rule: Change Request - Edit actions (API Enabled)

### I: Related record Actions

```
if (Status changed from: any value to: 'Closed' during record's last modification) {
    if (Spawning Problem ID!=NULL) {
        Set "CR Closed?" in Problem to Yes
    }
    if (Spawning SR ID!=NULL) {
        Set "CR Closed?" in Service Request to Yes
    }
}
```

### I: Assignee Change Notifications

```
if (Updated By='$assigned_person' and (Assigned Person changed from: any value to: any value last user's modification or
Assigned Team changed from: any value to: any value last user's modification)) {
    if (Assigned Person!=NULL) {
        Email Assigned Person of new assignment
    } else {
        Email Assigned Team of new assignment
    }
    if (Assigned Person changed from: any value to: any value last user's modification) {
        Increment Number of Assignees
    }
    if (Assigned Team changed from: any value to: any value last user's modification) {
        Increment Number of Teams Assigned
    }
}
```

## Rule: Parent/Child CR Handling

Runs on create and edit, including edits by other rules on creation

Saved Search: CR Type != Ind.

```
if (CR Type='Parent') {
    if (Status changed from: any value to: 'Approved' during record's last modification) {
        Approve linked Child CRs
    } else if (Status changed from: any value to: 'Rejected' during record's last modification) {
        Set status of child CRs to Rejected
    }
    if (CIs for Child CRs changed from: any value to: any value last user's modification) {
        Set Latest CR Field in Linked CI
    }
}
if (CR Type='Child' and Change Summary!~='child of') {
    Add Child of prefix to change summary
}
```

## FORMATTING FOR ACTION BUTTONS

### Add Time

**Purpose:** Provide users with a way to add Time Entries to a Change Request from the Change Request record itself.

**On Click:** Save Record, then execute action.

**Actions:**

#### I: Add Time

```
if (Time Spent=NULL) {
    Time spent must not be blank
}
Convert to Time Entry
Set Time Spent and Time Description to Blank
```

**Notes:** Triggers action to validate that Time Spent is not blank, then converts Time Spent and Time Description fields to a Time Entry record linked to the current Change Request record, then blanks out Time Spent and Time Description

### Set CI to Active

**Purpose:** Updates the linked CI to change its status to Active

**On Click:** Execute action

**Actions:**

#### U: Set Status of linked CI to Active

**Update Field:** Configuration_Item:Status

**Update Value:** Installed

### Set CI to In Maintenance

**Purpose:** Updates the linked CI to change its Status to In Maintenance

**On Click:** Execute Action

**Actions:**

#### U: Set linked CI to In Maintenance

**Update Field:** Configuration_Item:Status

**Update Value:** In Maintenance

### Spawn Child CR

**Purpose:** Converts the CR to multiple child CRs linked to the CI's selected in the CIs for Child CRs field.

**On Click:** Execute action

**Options:** Visible only when CR Type is Parent

**Actions:**

#### C: Spawn Child CR

**Conversion:** Spawn Child CR

**Options:** Allow user to reject creation, create record if does not exist, conversion runs interactively

## COMPANIES TABLE

## FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments |
|---|---|---|
| *Company Name | Short Text | Required field: Company must have a name |
| # of Employees | Integer | For tracking company size |
| Annual Revenue | Currency | For tracking company annual revenue |
| Assets | Linked table | Linked from Configuration Item, Company Name-> Client Company |
| Attached File(s) | File | For any associated files, such as company logo, marketing materials, etc. |
| Billing Address 1 | Short Text | |
| Billing Address 2 | Short Text | |
| Billing City | Short Text | |
| Billing Country | Choice | |
| Billing same as shipping | Multi-Choice | |
| Billing US State | Choice | |
| Billing US State Long | Choice | |
| Billing Zip or Postal Code | Short Text | |
| City | Short Text | |
| Company Name | Short Text | Required Field |
| Company Roles | Multi-Choice | Choose From: Customer, Prospect, Partner, Manufacturer, Vendor, Other<br>Default: Customer |
| Company Table | type | Type of the SW object |
| Contacts | Linked data from another table | Displays all records from the Contacts table linked to the individual company |
| Contract Info | Append Only Text | For providing notes and updates to the company contract information |
| Contract Status | Choice: Valid, Expired, No Incidents Remaining, No Contract | Defaults to No Contract<br>For tracking contract validity or renewal status |
| Contracts | Linked data from another table | Displays all records from the Contract table linked to the individual company. |
| Country | Choice | |
| Currency | Short Text | Defaults to US $ |
| Customer ID | Short Text | Used for tracking a customer number outside of Agiloft |
| Expiration Date | Date | For contract expiration date |
| Fax | Telephone/Fax | For company's fax line |
| Find local hotels | Action Button | Opens URL:<br>"http://maps.google.com/maps?f=q&q=hotels" + "+loc:+" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state |
| Incidents Remaining | Integer | Tracks the number of support incidents remaining |

100

| | | on a per-incident company contract before a renewal is needed. |
|---|---|---|
| Industry | Choice: List of Industries (see Lead table) | For tracking and grouping companies by associated industry |
| Intl State or Province | Short Text | |
| Main Contact | Compound | Combines Main Contact First Name and Main Contact Last Name |
| Main Contact Phone Main Contact Email Main Contact First Name Main Contact Last Name | Link to selected fields from other table | Links to People Table: Allows selection of a main point of contact at a company, from among Contact records designated as employees of that Company |
| Map It | Action Button | Opens URL: "http://maps.google.com/maps?f=q&q=" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state |
| Notes | Text | For general notes about the Company |
| Opportunities | Linked data from Opportunities table | Displays all opportunities linked to the particular Company record |
| Original Start Date | Date | Auto populates with the record's creation date |
| Parent Company | Link to single field from Company table: Company Name | Allows a link to another Company record for tracking parent/child company relationships |
| Primary Role | Choice | Choose From: Customer, Prospect, Partner, Manufacturer, Vendor, Other Default: Customer |
| Public Company | Choice: Yes/No | Tracks if this Company is publically traded |
| Purchase Orders | Linked data from another table | Links to PO Table: Company Name -> Linked Company |
| Quotes | Linked data from another table | Shows quotes related to this company record. |
| Reference as of | Date | Auto populates with the date that "Use as Reference" is set to "Yes" |
| Refresh User | Action Button | Saves record and reopens for edit to set most recent user to person editing record |
| Region | Choice: Western US, Central US, Eastern US, Canada, Europe, Asia, Africa, Other | Assists sales team in tracking company's time zone |
| Renewal Number | Integer | *Defaults to 0* Tracks the number of contract renewals, for customer longevity reporting |
| Rep Phone Rep Email Account Rep | Link to fields from Employee table: Direct Phone, Email, Full Name, | Links to the name, phone number, email, and team of the account rep associated with the Company record |

| Primary Team | Primary Team | |
|---|---|---|
| Required Response Time | Elapsed Time | For tracking SLA requirements, such as a response within 24 hours |
| Start Date | Date | Defaults to the creation date of the record |
| Secondary Contact | Compound | Combines Secondary Contact First Name and Secondary Contact Last Name |
| Secondary Contact Phone Secondary Contact Email Secondary Contact First Name Secondary Contact Last Name | Link to selected fields from another table | Links to People Table |
| Start Date | Date | |
| State or Province | Short Text | |
| Street Address 1 | Short Text | |
| Street Address 2 | Short Text | |
| Support Cases | Linked data from another table | Links to Support Case: Displays records from the Support Case table linked to the Company. |
| Support Hours | Choice: 24/7, Standard Business Hours, Special Hours | For tracking special support hour types and requirements |
| Telephone | Telephone/Fax | For Company's main phone line |
| Type of Support | Choice: None, Number of Incidents, Annual, Other | Defaults to None For tracking different types of support contracts |
| US State | Choice | |
| US State Long | Choice | |
| Use as Reference | Choice: Yes, No | Defaults to No For tracking Companies who have agreed to serve as references for your organization |
| Website URL | URL | For the website associated with the Company |
| Zip or Postal Code | Short Text | |

## Company Saved Searches

The default searches provided for this table are detailed below.

| Saved Search Name | Search Description |
|---|---|
| My Company | Returns records where the Company Name and Company City match those of the logged-in user |
| Reference Accounts | Shows all records for companies who have agreed to |
| Customers | Company Roles includes "Customer" |
| Manufacturers | Company Roles includes "Manufacturer" |

| Partners | Company Roles includes "Partner" |
|---|---|
| Prospects | Company Roles includes "Prospect" |
| Vendors | Company Roles includes "Vendor" |

## FORMATTING FOR ACTION BUTTONS

### Find local Hotels

**Purpose:** Automatically searches Google Maps for hotels near the company.

**On Click:** Opens URL: "http://maps.google.com/maps?f=q&q=hotels" + "+loc:+" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state

### Map it

**Purpose:** Automatically searches Google Maps for the location of the company.

**On Click:** Opens URL: "http://maps.google.com/maps?f=q&q=" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state

### Refresh User

**Purpose:** Saves record and reopens for edit to set most recent user to person editing record.

**On Click:** Save and Open record for edit

## CONFIGURATION ITEMS TABLE

### FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments |
|---|---|---|
| Annual Cost | Currency | |
| Attached Files | File | For attaching relevant files |
| Change Requests | Related table: Change Request | Displays all Change Requests related to this CI |
| CI Class | Choice: see Dependencies below | Dependent choices based on CI Type |
| CI Name | Short Text | Required, Unique<br><br>The name of the item |
| CI Tag | Short Text 25/20 | Unique, for managing bar code or other asset tags |

| *CI Type | Choice: Communication Device, Computer, Computer Peripheral, Infrastructure Item, Network Hardware, Printer, Server, Software Application, Storage Device | Required |
|---|---|---|
| Client Company | Link to selected fields in Company: Company Name | Only show if: Ownership = Client Company – this field is used when supporting client CI's as well as internal company CI's<br><br>Display Company using the "Company in other tables" view |
| Communications | Default Communications (Emails) table | |
| Contracts | Related Table: Contracts | |
| CPU | Short Text 20/30 | Only show if: Configuration Item Type = Server or Computer<br>For noting the hardware's CPU type |
| Date Installed | Date/Time | |
| Date Last Upgraded | Date | For tracking upgrade history |
| Date of Next Scheduled Service | Date/Time | Allows you to send upgrade task reminder notices |
| Date Purchased | Date | For calculating Configuration Item age |
| Date Stopped Working | Date/Time | Default value: Date the Operational Status field was last changed from "Working" to either "In Maintenance" or "Non-Operational"<br>Input box 10 chars wide |
| Date Warranty/Support Expires | Date | For tracking expiration of service contracts |
| Description | Short Text 90/255, 2 rows high | |
| Downstream CIs | Related Table: Configuration Item | Display Configuration Items using Summary view |
| Incidents | Related table: Incident | Displays all Incidents related to this CI, Embedded View |
| Invoice Number | Short Text 20/40 | |
| Latest CR Backout Plan, Latest CR Business Justification, | link to selected fields in Change Request table | This set is populated by a rule when a Child CR is edited and the CI's for Child CR's field is |

| | | |
|---|---|---|
| Latest Change Category, Latest CR Description, Latest Change Summary, Latest CR ID, CR Risk if Not Done, Latest CR Service Category, Latest CR Service Title, CR Urgency of Change | | changed.  It is used to generate Child CRs linked to the Parent CR and to populate the Parent CR fields using conversion |
| Manufacturer | Link to selected fields in Company: Company Name | Restrict values: Filter by: Saved search: Company Roles contains Manufacturer |
| Memory (Gb) | Choice: 1, 2, 4, 16, 32, 64, 128 | Only show if: Configuration Item Type = Server, Printer, Computer, or Communication Device |
| Model | Short Text 20/90 | Only show if: Configuration Item Type = Storage Device, Server, Printer, Network HW, Infrastructure Item, Computer Peripheral, Computer, Communication Device |
| Number of Licenses | Integer | *Defaults to 1* Useful for tracking per-seat licensed software Only show if: Configuration Item Type = Software Application |
| Operating System | Link to selected fields in Configuration Item: Configuration Item Name | Only show if: Configuration Item Type = Server or Computer Restrict values: Filter by: Saved search: Configuration Item Class = Operation System |
| Operational Status | Choice: Working, Non-Operational, In Maintenance | Default: Working |
| OS used in | Related table: Configuration Item (Link: Configuration Item Name -> Operating System) | Only show if Configuration Item Class = Operating System |
| Ownership | Choice: Company Configuration Item, Assigned to an Individual, Client Company Configuration Item | Default: Company Configuration Item Pop-up text: This field defines whether the Configuration Item is a general company Configuration Item, an internal Configuration Item |

| | | assigned to an individual, or an Configuration Item associated with an outside client company |
|---|---|---|
| PO Number | Short Text 20/30 | |
| Problems | Related table: Problems | Displays all Problems related to this CI<br>Lookup view, left justify |
| Purchase Cost | Currency | Purchase cost |
| Serial number | Short Text 35/255 | Note the Software's serial number |
| Service Requests | Related table: Service Requests | Displays all Service Requests related to this CI. Summary view, left justify. |
| Spawned CR Type | choice: CR Type values | This field has a default value of Child and is used when converting a CI record to a Child CR. It should not be deleted. |
| Status | wfstate | Default: Installed<br>See workflow diagram above |
| Team Leader<br>Responsible Team | Link to selected fields in Team:<br>Team Leader, Team Name | Default: Support Team<br>Restrict values: Filter by:<br>Saved search: Team Name contains Support Team or Staff Team<br>Display: Team Leader - Hyperlinked view only, Team Name - List of values |
| Type | Type of the SW object | |
| Type of License | Choice: Site license, Multiple users, Single user, Free license, Beta license, Other | Only show if: Configuration Item Type = Software Application |
| Upstream CIs | Link to selected fields in CI:<br>CI Class<br>CI Name<br>CI Tag<br>CI Type<br>Status | Allow multiple values, fast search. |
| User Department<br>User Phone<br>User Email<br>User Name<br>User Login<br>User Manager<br>User Team<br>User Title | Link to selected fields from People:<br> Department, Direct Phone, Email, Full Name, Primary Tem, Title | Tracks contact information for the user<br>Allow entries not in source table |

| | | |
|---|---|---|
| Vendor | Link to selected fields in Company: Company Name | Filter by Saved Search: Company Roles contains Vendor |
| Version | Short Text 10/25 | Only show if: Configuration Item Type = Software Application |
| Working Notes | Append-only text 90 chars, 6 lines | |
| zAll Contracts for This CI | Embedded Search Result | |
| zContracts Covering Just this CI | Linked table: Contracts | |
| zContracts that include this CI | Linked table: Contracts | |

## Asset Type and Class Dependencies

The tables below show the current setup of dependencies between CI Type and CI Class. These dependencies can be changed, as can all the choices, to meet your needs. The dependencies are also shown in the spreadsheet fields_and_choices.xls.

| CI Type values | CI Class (linked) values |
|---|---|
| Storage Device | Tape Library<br>Storage Switch<br>Storage Management Server<br>Storage Device<br>Storage Array |
| Software Application | Windows Service<br>Web Service<br>Web Server Application<br>Web Application<br>Other Application<br>Operating System<br>Infrastructure Software<br>Google Search Appliance<br>Desktop Application<br>Database Instance<br>Database Application<br>Application Server |
| Server | Virtual Machine<br>Storage Management Server<br>Java Server<br>FTP Server<br>Firewall<br>File Server<br>Email Server<br>Directory Server |

| Printer | Scanner |
| --- | --- |
| | Network Printer |
| | Label Maker |
| | Individual Printer |
| | Fax Machine |
| Network Hardware | WAN Interface |
| | VPN Appliance |
| | Voice Gateway |
| | Virtual Machine |
| | Other Hardware |
| | Network Switch |
| | Network Appliance |
| | Network Adapter |
| | Monitoring Device |
| | Load Balancer |
| | KVM Switch |
| | IP Switch |
| | IP Router |
| | IP Network |
| | IP Device |
| Infrastructure Item | Web Site |
| | Web Service |
| | VPN Appliance |
| | Voice Gateway |
| | Virtual Private Network |
| | Virtual Machine |
| | Video Conferencing |
| | Rack |
| | Monitoring Device |
| | Load Balancer |
| | FTP Server |
| | Firewall |
| | File Server |
| | Cluster |
| Computer Peripheral | Video Card |
| | UPS |
| | Peripheral - Other |
| | Other Hardware |
| | Mouse |
| | Monitor |
| | Modem |
| | Disk Drive |
| | Camera |
| | Bar Code Scanner |
| Computer | Windows Workstation |
| | Virtual Machine |
| | Laptop Computer |
| | Desktop Computer |

| Communication Device | Wi-Fi |
| --- | --- |
| | Voice Gateway |
| | Video Conferencing |
| | PDA enabled Cell Phone |
| | Modem |
| | IP Conference Phone |
| | Fax Machine |
| | Desktop Phone |
| | Cell Phone |

## RULES AND ACTIONS

## Create Child CR

Runs on Edit by API, Saved Search: R: Latest CR ID Changed

### I: Convert to Child CR

**Conversion:** Change Request

**Options:** Silently with no confirmation

## Edit: Actions triggered by web or API

**Purpose:** Adds assets to Contracts when Latest Contract ID added to.

**Runs:** Edit by Web or API

**Search Filter:** None

**Priority:** 11

**Actions:**

### I: Edit actions by api or web

```
if (Latest Contract ID changed from: any value to: any value during record's last modification) {
        Update linked Contract to add this asset
}
```

## SAVED SEARCHES

Default searches provided are detailed below.  The My Assigned search is used by the Inbox.

| Saved Search Name | Search Description |
| --- | --- |
| Computers by Name | CI Type='Computer' |
| F: In Stock CIs | Status='In Stock' |
| F: OS Applications | CI Class='Operating System' |

| My Configuration Items | User Login='system' |
|---|---|
| My Team CIs | |
| Installed CI's | Status = Installed |
| Responsible Team is in My Teams | As name suggests |
| Software Applications by Class | CI Type='Software Application' |

# CONTRACTS TABLE

## FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|
| Amendments | Related Table : Contracts | Conditional Visibility: Record Type != Amendments |
| Approval Action | Choice | Choices: Approved - Route Forward, Rejected - Route Back One Step, Rejected - Send Back for Resubmission. Visible only if Show Approval Steps has a Yes value. |
| Approval Notes | Append Only Text | Approver Notes Display size: 90x5 |
| Approvals Needed | Related Table : Approvals | Related table: filtered to those whose source is from Approval Template for a parallel workflow. |
| Approvals Received | Related Table : Approvals | Approvals filtered to those whose source is from the contract table. |
| Assigned Person - Not Used | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | |
| Assigned Team | Link to selected fields from other table:Teams Allow multiple values:No Allow entries not in source table:No | Strict link, filtered to those teams available for Contracts. Set by default to Contract Management Team. |

| | | |
|---|---|---|
| Cancel Contract | Action button | This action button will cancel the contract and send an email to the contract requester. The button is only visible to Admins and Contract Managers. |
| CI Involvement | Choice: No CI's, For Single CI, For Multiple CI's | Defines whether the contact is linked to one or more CI's, determines the visibility for the CI fields |
| CI's in this Contract | Link to selected fields from other table:Configuration Items Allow multiple values:Yes, fast search Allow entries not in source table:No | Visible only if CI Involvement= For Multiple CI's. |
| City, Company Name, Country, Company ID, Company Main Contact, Company Main Contact Email, Company Secondary Contact, Company Secondary Contact Email, State or Province, Street Address 1,Street Address 2,US State, Zip or Postal Code | Link to selected fields from other table:Companies Allow multiple values:No Allow entries not in source table:No | identifies the vendor or customer company linked to the contract |
| Communications | Communications Search Result | |
| Contract Amount | Currency | Monetary value of contract. Round to 2 decimal points; Display size: 12x1 with $ currency symbol; Always show 0 decimal digits. Once status passes into Pending Approval, the contract creator group can no longer edit this field. |
| Contract Description | Text | for details about the contract's purpose |
| Contract End Date | Date | |
| Contract Files | File | To manage drafts and final versions of the contract |
| Contract ID for Conversion | Link to single field from other table:Contracts Allow multiple values:No Allow entries not in source table:No | |

| Contract Owner Email, Internal Contract Owner, Contract Owner Login, Contract Owner Manager | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | Defines who is responsible for the contract. Filtered to people on teams related to contracts. Person who creates the contract by default, if created by staff user. If end users are going to be able to create contracts, this default value should be changed to none. |
|---|---|---|
| Contract Party Type | Choice | |
| Contract Party Types, Contract Types, Workflow Title, Approval Type | Link to selected fields from other table:Workflows Allow multiple values:No Allow entries not in source table:No | Strict link, filtered to those whose Contract Party Types and Contract Types are contained within the Contract values, and whose Used in is Contract. |
| Contract Start Date | Date | |
| Contract Term in Months | Integer | Default: 12 |
| Contract Title | Short Text | Brief name of contract |
| Contract Type | Choice | Product Support and Upgrades, Software License, Subscription Service, Parts only Contract, Service Contract, NDA, Partner Agreement, Other |
| Create and Attach Contract | Action button | On click, when Contract is for Product Support and Upgrades, runs an attached record action to create and Support Contract template . If the Contract is for Subscription Service or Service Contract, runs an attached record action to attach an SLA template; Visible only when Contract is for a Service Contract, Subscription Service or Product Support and Upgrades; Pop-up description: Use this button to create and attach a contract based on fields in this record |
| Create Related Contract | Action button | This button creates a new contract, mapping several of the fields and showing the new contract on the screen. When the new contract is saved, any assets linked to the old contract will be linked to the new one. Visible only when the Status is Renewed. |
| Create Renewal Contract | Action button | |
| Current Approver, Current | Link to selected fields | Strict link, always update; default value is |

| Approval Team, Current Step ID, Next Approver, Next Approval Team, Next Step ID, Previous Approver, Previous Approval Team, Previous Step ID, Progress Bar Image, Step Number, Current Step Title | from other table:Steps Allow multiple values:No Allow entries not in source table:No | the first step of the linked workflow |
|---|---|---|
| Days in advance to notify for renewal | Integer | Default: 45 |
| Has Related Contracts | Choice | Determines the visibility of the related contracts table. |
| Has Related Subcontracts | Choice | This controls visibility of the Subcontracts table. |
| Inbound Contract Files | File with Versioning | For files sent from a vendor or party to the contract during the approval process. |
| Insurance Certificates | Related Table : Insurance Certificates | |
| Is the new contract a Renewal? | Choice | This is mapped in conversion to the Renewal Contract: Yes/No field in the new contract so we know whether to maintain the link to the previous contract id or blank it out. |
| Mark as Signed | Action button | Sets status to Active or Signed depending on if the Contract Start Date has passed or not. |
| New Record Type | Choice | Same choices as for Record Type: Master Contract, Contract, Subcontract, Amendment. This field is used when using the action button to convert to a new contract record. Choosing the type of new record here sets the record type of the new record. Dependent Choices are set up based on the Record Type field. |
| Notes To/From Party Contacts | Append Only Text | Putting notes in this field and clicking the Send to Party Contacts button will send an email with the notes and the attached contract files to the Primary and Secondary contact email addresses. Replies by email will go back into this field. |
| Number of Approvals Needed | Calculation on Multiple Linked | Count of approvals needed in all statuses |

| | Records : Approvals | |
|---|---|---|
| Number of Approvals Received | Calculation on Multiple Linked Records : Approvals | Count of records filtered to those whose Approval Status is Approved |
| Number of Related Contracts | Calculation on Multiple Linked Records | Used by rule to determine visibility of Related Contracts section on layout |
| Number of Subcontracts | Calculation on Multiple Linked Records | Used by rule to determine visibility of Subcontracts section on layout |
| Number of times sent for approval | Integer | This field will be used to determine if the approval cycle has begun. The field will be incremented when the Submit for Approval button is clicked. It will also be used to determine in parallel approvals if approval records should be created. |
| Number of Valid Insurance Certificates | Calculation on Multiple Linked Records : Insurance Certificates | Sum of the Insurance Certificates that are in a Status of Valid i.e. Filtered to those whose expiration date has not passed and are valid |
| Ongoing Notes | Append Only Text | Used to hold information about the contract that does not relate to approval or renewal. Visible to contract staff users and contract end users, but not vendors. |
| Parent Contract Amount, Parent Contract End Date, Parent Contract Start Date, Parent Contract Title, Parent Contract ID, Parent Contract Record | Link to selected fields from other table:Contracts Allow multiple values:No Allow entries not in source table:No | This is the control field for many other conditionally visible fields in this table, including amendments, subcontracts, etc. Mapping: Strict Link Update: Automatically in the background Filtered: To contracts with a Record Type of not Amendment. Conditional Visibility: Record Type == Amendment or Subcontract |
| Previous Contract End Date, Previous Contract Title, Previous Contract ID, Previous Parent Contract ID | Link to selected fields from other table:Contracts Allow multiple values:No Allow entries not in source table:No | Populated automatically with the source contract when a contract is created using the action button in the previous contract to create the renewal contract. This field is used by a rule to get to the assets for the old contract and update them into the new contract. Visible only if "Is this Contract a Renewal" field is Yes, otherwise hidden. |
| Record Type | Choice | Choices: Master Agreement, Contract, |

| | | Subcontract, Amendment. Default: Stand-Alone Contract |
|---|---|---|
| Related Contracts | Related Table : Contracts | |
| Renewal Contract | Choice | Default: No, set by conversion to Yes when a new contract is created from the action button and the user has selected Yes in the Is the New Contract a Renewal field. Controls rule running on new contract creation. |
| Renewal Contract Start Date, Renewal Contract Title, Renewal Contract ID | Link to selected fields from other table:Contracts Allow multiple values:No Allow entries not in source table:No | Conditionally visibility: Field displayed when Status != Draft OR Pending Approval. |
| Renewal Cost | Currency | Monetary cost to renew contract. Round to 2 decimal points; Display size: 12x1 with $ currency symbol; Always show 0 decimal digits |
| Renewal Email Sent on | Date | |
| Renewal Notes | Append Only Text | |
| Renewal Notification Date | Date | set to Contract End Date minus 45 days, visible only if Renewal Type= Notify Staff to Renew |
| Renewal Status | Choice | Renewal Email Sent, Declined Renewal, Unresponsive, Renewal Quote Accepted, Invoiced, Paid |
| Renewal Type | Choice | Automatically renews, Notify staff to renew, One time contract, default is Notify Staff to Renew |
| Requester Phone, Requester Email, Requester Name, Requester ID | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | Strict link, to the person who has requested the contract and should be notified about it. |
| Review Completed | Action button | Depending on the Approval Action, the button either marks the Contract as approved, sets it to the next approval |

| | | step, or back to the preceding or first step, and sends notifications to the contract owner (in case of rejection) or to the next assigned team otherwise. |
|---|---|---|
| Send Back for Reapproval | Action button | Validates that notes are added, then sets Step ID and Status to Start Step ID and Pending Approval or Requires Reapproval, as necessary. Emails Current Approval Team of reapproval. |
| Send Contract to Party | Action button | Validates that notes are provided to Party, and that party email exists, then emails notes to party |
| Sent to Party By | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | Person who clicks the Send to Contract Party button. Updated by the action button. Used to know who to notify if the contract party replies. |
| Show Approval Steps | Choice | This field is set by a rule to Yes when the status is Pending Approval and when the workflow is sequential. Otherwise it is set to No. It is the parent for visibility of the Current Step fields and the workflow graphic and the Approval Action and Review Completed Button. |
| Status | Choice | |
| Subcontracts | Related Table : Contracts | Conditional Visibility: Record Type == Master Contract, filtered to records whose Record Type = Subcontract. |
| Submit for Approval | Action button | On click, sets the User Status to Pending Approval then saves and closes the record; Visible only if Status equals Draft |
| Submit for Review | Action button | Emails Contract Manager Team of New Contract, then sets Status to Pending Contract Manager |
| Times Sent for Approval | Integer | Integer - this is incremented each time someone hits the Submit for Approval button and keeps track of how many times a contract is sent back to the beginning. |
| Total Number of Steps, Starting | Link to selected fields from other | A second set from workflow that is visible only if approval type is sequential |

| Step ID | table:Workflows<br>Allow multiple<br>values:No<br>Allow entries not in<br>source table:No | so it can be hidden for the parallel<br>workflows. Default: same workflow title<br>as the main set of fields. |
|---|---|---|

## RULES AND ACTIONS

The Contracts Table has the following rules set up. Rules which are run based on a schedule (rather than those which are event-triggered) are identified by "TB", for "Time based".

### Create: All New Contract Actions

**Purpose:** This rule handles all update and linking actions upon creating a new record.

**Runs:** Upon creation by Web or API

**Search Filter:** None

**Priority:** 1

**Actions:**

#### I: All New Contract Actions

```
if (Contract Start Date!=NULL and Contract End Date!=NULL and Contract End Date<'$contract_start_date') {
    contract end date can't be less than start date
}
if (Status='Signed' and Contract End Date>1 day in the future and Contract Start Date>1 minute old) {
    Set Status to Active
}
if (Renewal Contract ID!=NULL and Renewal Contract='No') {
    Blank out Previous Contract ID
}
if (Renewal Contract='Yes' and Previous Parent Contract ID=NULL) {
    Blank out Parent Contract ID
}
if (Renewal Contract='Yes' and Previous Parent Contract ID!=NULL) {
    Copy Previous Parent Contract ID to Parent Contract ID
}
if (Previous Contract ID!=NULL and Renewal Contract='Yes') {
    if (CI Involvement='For One or More CI's') {
        Copy contract id to latest contract id in assets
    }
    Copy contract id to previous contract Renewal Contract Id
}
```

#### U: Update Renewal Notification Date

**Updated Field:** Renewal Notification Date

**Updated Value:** $contract_end_date – ($days_in_advance_to_notify_for_renewal) DAYS

117

## Edit: Updates by Party

**Purpose:** This rule handles notifications for Contracts that are updated by an outside party that does not have direct access to the system.

**Runs:** Upon edits by Email or Web

**Search Filter:** R: Updates by Vendor or Customer Team

**Priority:** 1

**Actions:**

### E: Email Owner and Sender of Outside Party Update

**Sent To:** Contract Manager Team, Contract Manager, Sent to Party By.

**Summary:** This email notifies the relevant users that an outside party has updated the contract.

## Edit: All Edit Actions by Web or API

**Purpose:** This rule handles the processes resulting from changes made by staff or rules.

**Runs:** Upon edits by Web or API

**Search Filter:** None

**Priority:** 2

**Actions:**

### I: All Edit Actions with API

```
if (Approval Type='Parallel' and Number of Approvals Needed='$number_of_approvals_received' and Number of Approvals
Needed>=1 and Status='Pending Approval') {
    Set Status to Approved
    Email Contract Owner of approved contract
}
if (Status changed from: any value to: 'Pending Approval' during record's last modification and Approval Type='Sequential'
and Show Approval Steps='No') {
    Set Show Approval Fields to Yes
} else if (Status changed from: 'Pending Approval' to: any value last user's modification and Show Approval Steps='Yes') {
    Set Show Approval Steps to No
} else if (Status changed from: any value to: 'Signed' last user's modification and Contract End Date>1 day in the future and
Contract Start Date>=1 minute old) {
    Set Status to Active
} else if (Status changed from: 'Cancelled' to: 'Pending Contract Manager' last user's modification) {
    Set Status of existing approvals to Requires Reapproval
    Clear Approval Action
    if (Current Step ID!='$starting_step_id') {
        Set Current Step ID to Start Step ID
    }
}
if (Times Sent for Approval>=1 and Workflow Title changed from: any value to: any value last user's modification) {
    Not Allowed to Change Workflows
}
if (Contract End Date changed from: any value to: any value last user's modification or Days in advance to notify for renewal
changed last user's modification) {
    Update Renewal Notification Date
}
if (Number of Subcontracts>=1 and Has Related Subcontracts='No') {
    Set Has Subcontracts to Yes
}
if (Number of Related Contracts>=1 and Has Related Contracts='No') {
    Set Has Related Contracts to Yes
}
```

## TB: Notify of upcoming expirations (Disabled by Default)

**Purpose:** This rule controls the email notifications for upcoming expirations of contracts

**Runs:** At selected time intervals; every 2 Days at 3 AM

**Search Filter:** R: Renewal date is tomorrow

**Priority:** 0

**Action:**

### E: Email Contract Owner about Renewal Date

**Sent To:** User in the Contract Manager field

**Summary:** Reminds the Contract Manager that the impending renewal date of a contract

## TB: Daily Check for Start Date (Disabled by Default)

**Purpose:** This rule checks for contracts whose start date has arrived and changes the Status accordingly

**Runs:** At selected time intervals; every 2 days at 4 AM

**Search Filter:** R: Status is signed and Start Date is today or earlier

**Priority:** 4

### I: Actions When Start Date Arrives

```
Set Status to Active
if (Previous Contract ID!=NULL) {
        Set Status of Previous Contract to Renewed
}
```

## TB: Daily Check for Expiration Date (disabled)

**Purpose:** This rule checks for contracts whose end date has arrived and takes the appropriate actions.

**Runs:** At selected time intervals; every 2 days at 2 AM

**Search Filter:** R: Contract Expired Today

**Priority:** 5

### I: Handle Expired Contract

```
if (Renewal Type='Automatically renews' and Status!='Renewed' and Status!='Cancelled') {
        Set Status to Renewed
        Convert to create new contract
} else {
        Set Status to Expired
}
if (Renewal Type='Notify staff to renew') {
        Email owner contract end date has passed
}
```

## FORMATTING FOR ACTION BUTTONS

## Cancel Contract

**Purpose:** This action button will cancel the contract and send an email to the contract requester. The button is only visible to Admins, Configuration Managers, Contract Owners, Service Managers, and Contract Managers.

**On Click:** Save record then execute actions.

**Visibility:** Only when Status is Active, Approved, Draft, Expired, Pending Approval, Renewed, or Signed.

**Actions:**

### I: Validations for Cancellations and Reapprovals

```
if (Approval Notes not changed last user's modification) {
        must add notes when rejecting
}
```

### U: Set Status to Cancelled

**Updated Field:** Status

120

**Updated Value**: Cancelled

### E: Email Requestor Contract Cancelled

**Email Template:** Contract has been cancelled - Requester

## Create and Attach Contract

**Purpose:** On click, when Contract is for Product Support and Upgrades, runs an attached record action to create and Support Contract template . If the Contract is for Subscription Service or Service Contract, runs an attached record action to attach an SLA template; Visible only when Contract is for a Service Contract, Subscription Service or Product Support and Upgrades; Pop-up description: Use this button to create and attach a contract based on fields in this record.

**On Click:** Execute action

**Visibility:** Only when Contract Type is Product Support and Upgrades, Service Contract, or Subscription Service.

### I: Create contract based on type of contract

```
if (Contract Type='Product Support and Upgrades') {
      Attach Support Contract
} else if (Contract Type='Subscription Service' or Contract Type='Service Contract') {
      Attach SLA
}
```

**Action Specific Items:** Attach Support Contract and Attach SLA work as advertised.

## Create Related Contract

**Purpose:** This button creates a new contract, mapping several of the fields and showing the new contract on the screen. When the new contract is saved, any assets linked to the old contract will be linked to the new one. Visible only when the Status is Renewed.

**On Click:** Save record, then execute actions.

**Actions:**

### I: New Contract Validations

```
if (Record Type='Master Agreement' and New Record Type='Master Agreement' or Record Type='Contract' and New Record
Type='Contract' or Record Type='Subcontract' and New Record Type='Subcontract' or Record Type='Amendment' and New
Record Type='Amendment') {
      Cannot Create a New Record of this Type
}
```

### C: Convert to new contract

**Conversion:** Contract

**Options:** Interactive, showing user new record page

## Create Renewal Contract

**Purpose:** Validates that another renewal contract doesn't already exist, sets certain fields correctly, then converts the current contract to create a new contract, then sets Is the New Contract a Renewal to No.

**On Click:** Execute actions.

**Visibility:** Status is Active, Expired, or Signed

### I: Contract Renewal Validations

```
if (Renewal Contract ID!=NULL) {
        Cannot Create a Second Renewal
}
```

### I: Contract Renewal Actions

```
if (New Record Type!='$record_type') {
        Set New Record Type to Match Contract Record Type
}
if (Is the new contract a Renewal?='No') {
        Set Is the New Contract a Renewal to Yes
}
```

### C: Convert to create new contract

**Conversion:** Contract

**Options:** Interactive, showing user the new record page

### U: Set Is the New Contract a Renewal to No

**Updated Field:** Is the new contract a Renewal?

**Updated Value:** No

## Mark As Signed

**Purpose:** Sets status to Active or Signed depending on if the Contract Start Date has passed or not.

**On Click:** Execute action.

**Visibility:** Status is Approved.

**Actions:**

### I: Contract Signing Actions

```
if (Contract Start Date<=this day) {
        Set Status to Active
} else {
        Set Status to Signed
}
```

## Review Completed

**Purpose:** Depending on the Approval Action, the button either marks the Contract as approved, sets it to the next approval step, or back to the preceding or first step, and sends notifications to the contract owner (in case of rejection) or to the next assigned team otherwise.

**On Click:** Save record, then execute actions.

**Visibility:** Show Approval Steps is Yes.

**Actions:**

### I: Validations for review completed

```
if ((Approval Action='Rejected - Route Back One Step' or Approval Action='Rejected - Send Back for Resubmission') and
Approval Notes not changed last user's modification) {
      must add notes when rejecting
}
if (Current Approval Team!<<'Company Team,Internal Customer Team,Customer Team,Marketing Team,1st Level Support
Team,Sales Team,2nd Level Support Team,Admin Team,Professional Services Team,Change Management Team,Change
Approver Team,Configuration Management Team,Network Operations Team,Server Team,Custom Applications
Team,System Administration Team,Backup and Storage Team,Database Team,Security Team,Purchasing Team,Desktop
Applications Team,Contract Management Team,Vendor Management Team,HR Team,Facilities Team,Document Services
Team,Office Mgmt Team,Knowledge Team,Vendor Team,Legal Team,Risk Team,Compliance Team,Finance Team,Contract
Owner Team') {
      you are not on approval team
} else if (Status!='Pending Approval') {
      cannot approve while in draft status
}
```

Note: Rule checking against $GLOBAL.my_teams

### C: Create Approval Record

**Conversion:** Approval

**Options:** Silently with no confirmation, don't report errors.

### I: Handle Review Complete Actions

```
if (Approval Action='NULL') {
    Must select an approval action
} else if (Approval Action='Approved - Route Forward') {
    if (Step Number='$total_number_of_steps') {
        Set Status to Approved
    } else {
        Set Next Step ID and clear approval action
        Email Current Approval Team ready for approval
    }
} else if (Approval Action='Rejected - Route Back One Step') {
    if (Previous Step ID=NULL) {
        Set Status back to Draft and clear approval action
        Email Assigned Team and Contract Owner of rejection
    } else {
        Set back to previous step and clear approval
        Email Current Assigned Team of Route Back
    }
} else if (Approval Action='Rejected - Send Back for Resubmission') {
    Set Status back to Draft and clear approval action
    Email Assigned Team and Contract Owner of rejection
    Set Status of existing approvals to Requires Reapproval
    if (Current Step ID!='$starting_step_id') {
        Set Current Step ID to Start Step ID
    }
}
```

## Send Back for Reapproval

**Purpose:** Validates that notes are added, and then sets Step ID and Status to Start Step ID and Pending Approval or Requires Reapproval, as necessary. Emails Current Approval Team of reapproval.

**On Click:** Saves record then executes actions.

**Visibility:** Status is Approved.

**Actions:**

### I: Validations for Cancellations and Reapprovals

```
if (Approval Notes not changed last user's modification) {
    must add notes when rejecting
}
```

### I: Send Back for Reapproval Actions

```
if (Approval Type='Sequential') {
      Clear Approval Action
      if (Current Step ID!='$starting_step_id') {
            Set Current Step ID to Start Step ID
      }
      Email Current Approval Team of reapproval
      Set Status of existing approvals to Requires Reapproval
} else if (Approval Type='Parallel') {
      Set linked approvals to Pending Approval
}
Set Status to Pending Approval
```

## Send Contract to Party

**Purpose:** Validates that notes are provided to Party then emails notes to party. If party email does not exist, generates a popup that warns the user that the email was not sent because there was no address.

**On Click:** Save record, execute actions, then save again.

**Actions:**

### I: Send to Party Actions

```
if (Notes To/From Party Contacts changed from: any value to: any value last user's modification and Company Main Contact
Email!=NULL) {
      Email Notes to Contract Party
      Set Sent to Party By field
} else if (Notes To/From Party Contacts not changed last user's modification) {
      Must provide content in Party Notes
}
if (Company Main Contact Email=NULL) {
      Warn that company email was blank
}
```

## Submit for Approval

**Purpose:** On click, sets the User Status to Pending Approval then saves and closes the record; Visible only if Status equals Draft.

**On Click:** Saves record and executes action.

**Visibility:** Status is Draft or Pending Contract Manager.

**Actions:**

### I: Run Submit for Approval Actions

```
Set Status to Pending Approval
Increment Times Sent for Approval by 1
if (Approval Type='Sequential') {
        Set Show Approval Fields to Yes
        Set Current Step ID to Start Step ID
        Email Current Approval Team ready for approval
} else if (Approval Type='Parallel') {
        if (Times Sent for Approval=0) {
                Copy Contract ID to approval templates
        } else {
                Set Status of all approvals to Pending Approval
        }
}
```

## Submit for Review

**Purpose:** Emails Contract Manager Team of New Contract, then sets Status to Pending Contract Manager.

**On Click:** Saves record, executes, actions, then saves record again.

**Visibility:** Status is Cancelled or Draft.

**Actions:**

### E: Email Contract Manager Team of New Contract

**Email Template:** Contract has been submitted for review.

### U: Set Status to Pending Contract Manager

**Updated Field:** Status

**Updated Value:** Pending Contract Manager

# DOCUMENTS TABLE

## FIELDS

All fields to be added to the table are indicated below, as well as any changes to the default fields.

*Required fields are denoted by an \* next to the field label*

| Field Label | Data Type/ max length/ display length | Comments / Default Value |
|---|---|---|
| Approval Notes | Append Only Text | Approval comments from linked approval records will be pushed to this field. |
| Approvals | Related Table | Table of all of the approvals generated for this document based on Potential Reviewers |

| | | |
|---|---|---|
| Assigned Person | Link to selected fields from other table: People<br>Allow multiple values: No<br>Allow entries not in source table: No | Display as List of Values; filtered to members of the Assigned Team |
| Assigned Team | Link to selected fields from other table: Team<br>Allow multiple values: No<br>Allow entries not in source table: No | Default: Document Management Team |
| Audience | Multi-Choice | Multi-choice: Customers, Partners, Prospects, Employees, Vendors, Executives, Media, Analysts; Display as checkboxes |
| Cancel | Action button | This action button when clicked changes the Status field to Cancelled. |
| Date Published | Date/Time | Date the status changed to Published. Date and Time box with Calendar and time pop-up. |
| Date Retired | Date | |
| Description | Text | For records with files attached, this could be a simple abstract. For FAQs this could be a more detailed description/solution. Maximum size: 5000; Display size: 90x5 |
| Document Access | Choice | This is used in group permissions to determine visibility. Choices: Public, Partners, All Internal Users, Restricted |
| Document Subtype | Choice | Subtype, values are dependent on the Type chosen. Choices: Newsletter, Landing Page, Demo Video, White Paper, Case Study, How to Guide, Product Datasheet, Press Release (visible for marketing collateral), Product User Manual, Desktop Apps Reference, HR Document, Corporate Record, Employee Communication, etc. |
| Document Type | Choice | Choices: FAQ, Memo, Policy/Procedure, Marketing Collateral, Product Documentation/User Manual, Other Reference |
| Final Due Date | Date | Calendar box with pop-up. |

| For Department, Department Manager | Link to selected fields from other table: Departments Allow multiple values: No Allow entries not in source table: No | Could be used to auto-populate the manager in the Potential Reviewers. Display 'For Department' as a list. Default is same as Submitter Department. |
|---|---|---|
| Internal File URL | Short Text | Used to indicate where a local may be located on a server. Maximum size: 255; Display size: 90x1 |
| Mark Retired | Action button | Handles state transition from published to retired. Visible only when State is Published. |
| Next Milestone Date | Date | Calendar box with pop-up |
| Number of Potential Reviewers | Calculation on Multiple Linked Records | Helper field for automation. Not displayed on Layout. |
| Priority | Choice | Choices: Urgent, High, Medium, Low |
| Publish | Action button | Sets status to Published. Visible only when status is Ready for Publication |
| Publish without Approval | Action button | Sets Status to Published. Visible only if Status is Pending Review; requires validation of Requires Approval = No. |
| Published Files | File with Versioning | Versioning enabled; checkin and checkout implemented and fully enforced. |
| Published Format | Choice | Choices: HTML, PDF, Powerpoint (.pptx), MS Word (.docx), MS Publisher (.pub), Visio, Adobe In Design, Other |
| Purpose | Text | Maximum size: 500; Display size: 90x5 |
| Related Document(s) | Link to single field from other table: Documents Allow multiple values: Yes, fast search Allow entries not in source table: No | A multi-select field to display all documents in the Documents table related to the current document; the current document will not be included; display as multiple value box with popup selection list |
| Requires Approval | Choice | Choices: Yes, No; Default value: Yes; Required |
| Return to Submitter | Action Button | Sets status to Draft and notifies submitter that their document needs review. |

| Reviewer(s) | Link to single field from other table: Employee<br>Allow multiple values: Yes, fast search<br>Allow entries not in source table: No | This is a multi-select field where you can either define an approval team of one or more potential reviewers. Used in the conversion process to create approval records. Filtered to display only members of Document Reviewers Team; Visible only if Requires Approval is set to Yes; display as multiple value box with popup selection list |
|---|---|---|
| Rush | Choice | Choices: Yes, No |
| Source Files | File with Versioning | Versioning enabled; Check in/out implemented and fully enforced; changes to document displayed in Source Document Changes field |
| Status | Choice | Workflow Choices: Draft, Pending Review, Pending Approval, Ready for Publication, Published, Retired, Cancelled |
| Submit for Approval | Action Button | Runs an action to set the Status to Pending Approval and initiate the conversion process. Require value in either Potential Approvers validation. |
| Submit for Review | Action button | Runs an action to set the Status to Submitted for Review. Visible only if 'Status' = Draft |
| Submitter, Submitter Department | Link to selected fields from other table: Employees<br>Allow multiple values: No<br>Allow entries not in source table: No | Default to creator |
| Title | Short Text | Required; Maximum size: 255; Display size: 90x1 |
| Total Number of Approvals | Calculation on Multiple Linked Records: Approvals | Count of Approval Records |
| Total Number of Approvals & Rejections | Calculation on Multiple Linked Records: Approvals | Filtered to Approvals with an Approval Status of Approved or Rejected |
| Total Number of Rejections | Calculation on Multiple Linked Records: Approvals | Filtered to Approvals with an Approval Status of Rejected |
| Total Number of Requires Reapprovals | Calculation on Multiple Linked Records: Approvals | Filtered to Approvals with an Approval Status of Requires Reapproval |
| Working Notes | Append Only Text | Display size: 90x5 |

## RULES

## Create: All Creation Actions

**Purpose**: This handles status changes and notifications to authors and publishers of documents.

**Runs**:  When created or edited by web, email or API

**Search filter**: None

**Priority**: 1

**Actions**:

### I: All Creation Actions

```
if (Status='Pending Review') {
     Email Document Mgmt Document Pend Review
}
```

## Edit: All Edit Actions without API

**Purpose**: This handles the process to begin the conversion action of Employee to Approvals.

**Runs**:  When edited by web or email

**Search filter**: None

**Priority**: 1

**Actions**:

### I: All Edit Actions

```
if (Status changed from: 'Pending Approval' to: 'Draft' last user's modification) {
     Email Submitter Document Requires Update
} else if (Status changed from: any value to: 'Published' last user's modification) {
     Email Submitter Document Published
} else if (Status changed from: any value to: 'Pending Review' last user's modification) {
     Email Document Mgmt Document Pend Review
} else if (Status changed from: any value to: 'Ready for Publication' last user's modification) {
     Email Submitter Document Ready for Publication
}
if (Reviewer(s) changed from: any value to: any value last user's modification and Reviewer(s)!=NULL and Status='Pending Approval') {
     Update Last Document ID in Employee
}
if (Status changed from: 'Pending Review' to: 'Pending Approval' last user's modification and Total Number of Requires Reapproval>=1) {
     Update Linked Approval Records to Pend Approval
}
```

## Edit: Approval Updates (API enabled)

**Purpose**: This handles any actions that should occur as a result of approvals and rejections.

**Runs**:  When edited by web, email or API

**Search filter**: R: Total Number of Approvals or Rejections Changed

**Priority**: 5

**Actions**:

### I: All Approval Actions

```
if (Total Number of Approvals & Rejections='$total_number_of_approvals') {
    if (Total Number of Rejections>=1) {
        Set Status to Draft
        Email Submitter Document Requires Update
        Set Approval Records to Requires Reapproval
    } else {
        Set Status to Ready for Publication
        Email Submitter Document Ready for Publication
        Email Doc Mgmt Team Document Ready for Pub
    }
}
```

## Edit: All Edit Validations

**Purpose**: This handles any validations for record edits.

**Runs**:  When edited by web or email

**Search filter**: None

**Priority**: 5

**Actions**:

### I: All Edit Validations

```
if (Status='Pending Approval' and Requires Approval='Yes' and Number of Potential Reviewers<1) {
    Must Provide Potential Approvers
}
```

## FORMATTING FOR ACTION BUTTONS

### Submit for Review

**Purpose**: This button will change the status of the record to Pending Review and save the record

**Visibility**: Only visible when Status is Draft or Retired

### Submit for Approval

**Purpose**: This button will change the status of the record to Pending Approval following validation, set Last Document ID field in Employee record and save the record.

**Visibility**: Only visible when Status is Pending Review

### Cancel

**Purpose**: This button will change the status of the record to Cancelled and save the record

**Visibility**: Only visible when Status is Draft

### Publish

**Purpose**: This button will change the status of the record to Published and save the record

**Visibility**: Only visible when Status is Ready for Publication

### Publish without Approval

**Purpose**: This button will change the status of the record to Published following validation and save the record

**Visibility**: Only visible when Status is Pending Review

### Mark Retired

**Purpose:** This button will change the status of the record to Retired and save the record

**Visibility:** Only visible when Status is Published

### Return to Submitter

**Purpose:** This button will change the status of the record to Draft, email the submitter to update the document and save the record

**Visibility:** Only visible when Status is Pending Review

## EUI TEMPLATES TABLE

### FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments |
|---|---|---|
| *Name | Short Text | The template name |
| Body | Text | This field holds the actual HTML template for the EUI |
| Created By Creator Name Creator Login Creator Team | Link to fields from Contact table: Full Name, Full Name, Login, Primary Team | Tracks creator information |
| Date Created | Date/Time | Auto populates at record creation |
| Date Updated | Date/Time | Automatically updated when the record is edited |
| Description | Text | A short description of the RUI template |
| ID | Auto-Increment | Auto incremented by the system, this is the record's unique ID number |

| Owned By | Link to single field from Contact table: Login | Tracks owner info |
|---|---|---|
| Updated By | Link to single field from Contact table: Full Name | Automatically updated when the record is edited |

## INCIDENTS TABLE

### FIELDS

*Required fields are denoted by an * next to the field label*

*Default fields are marked in grey*

| Field label | Data Type | Admin notes |
|---|---|---|
| Add Time | Action button | Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the current Change Request record, then blanks out Time Spent and Time Description |
| Add to Knowledgebase | Choice | |
| Additional Information | Append Only Text | For communicating with customer and allowing the customer to provide updates |
| All Time Entries | Related Table : Time Entries | |
| All Time Spent | Calculation on Multiple Linked Records : Time Entries | |
| Assign to CI Responsible Team | Choice | Only show if Problem CI Identified? = Yes |
| Assigned Person, Assignee Manager | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | filtered to members of assigned team |
| Assigned Team Manager, Assigned Team, Team Pager | Link to selected fields from other table:Teams Allow multiple values:No Allow entries not in source table:No | |
| Attached Files | File with Versioning | |
| Change Request ID, Linked Problem ID, Problem Status, Problem Summary | Link to selected fields from other table:Problems | |

| | Allow multiple values:No Allow entries not in source table:No | |
|---|---|---|
| CI Class, CI Name, CI Tag, CI Type, CI Responsible Team, CI Serial number, CI Status | Link to selected fields from other table:Configuration Items Allow multiple values:No Allow entries not in source table:No | Required |
| Closed By, Closed By Team | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | |
| Communications | Communications Search Result | |
| Convert to Change Request | Action button | Save and Close record and convert to create a new Change Request |
| Convert to Problem | Action button | Save before and after conversion, prevent conversion if the problem ID already has a value. |
| Convert to Service Request | Action button | Converts silently |
| CR Change Category, CR Summary, CR ID, Change Request Status, CR Urgency | Link to selected fields from other table:Change Requests Allow multiple values:No Allow entries not in source table:No | |
| Date Closed | Date/Time | |
| Description | Text | |
| External CCs | Short Text | |
| history | History | |
| I would like to reopen my Incident | Choice | Defaults to No This is used to flag an Incident for reopening and further action |
| Impact | Choice | Number or percentage of affected users Required |
| Incident Reported via | Choice | Required Default: Web |
| Internal CCs | Link to: Employees | Single field link with multiple values |

| | | enabled. |
|---|---|---|
| Knowledgebase Topics | Multi-Choice | Required<br><br>Choose from:<br><br>Software Troubleshooting, Printing Troubleshooting, Network/Access Troubleshooting, PC/Laptop Troubleshooting, Phone/Cell Troubleshooting, Other |
| Number of Assignees | Floating Point | |
| Number of Reopens | Floating Point | |
| Number of Teams Assigned | Floating Point | |
| Priority | Choice | |
| Problem CI Identified? | Choice | Required |
| Problem Closed? | Choice | Flag for notification when linked Problem closes. Not shown to users.<br>Default: No |
| problem conversion map | Choice | |
| Publish to Knowledgebase | Choice | |
| Resolution | Text, 10000 90 x 6 | For providing a solution before closing an Incident |
| Service Request ID, Service Request Status, Service Request Summary | Link to selected fields from other table:Service Requests<br>Allow multiple values:No<br>Allow entries not in source table:No | |
| Set Priority | Action button | Automatically sets priority based on impact and urgency. |
| Solution Files | File | |
| Spawning SR ID | Link to selected fields from other table:Service Requests<br>Allow multiple values:No<br>Allow entries not in source table:No | For conversion linking, will hold the ID of the Service Request record that was converted to this Incident |
| Staff Only Notes | Append Only Text | For communicating internally – this field is not visible to end users |
| Standard Solution | Choice | |

| Status | Choice | Choose From: Open, Assigned, Closed, Sent to Customer, Updated by Customer, Workaround Provided, Reopened |
|---|---|---|
| | | These choices represent states in the workflow field |
| Submitter Department, Customer Phone, Customer Email, Customer Name, Customer Login, Customer Team | Link to selected fields from other table:People Allow multiple values:No Allow entries not in source table:No | Required |
| Summary | Short Text | Required |
| Time Description | Text | |
| Time Spent | Elapsed Time | |
| Total Hours to Close | Elapsed Time | |
| Type of Problem | Choice | |
| Urgency | Choice | Required |
| Working Hours to Close | Elapsed Time | |

# RULES

## Set Priority on New Tickets

Saved Search: R: Priority is Empty

### I: Set Priority

```
if (Impact='Affects Single User') {
    if (Urgency='Low') {
        Set Priority to Low
    } else if (Urgency='Medium') {
        Set Priority to Medium
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to High
    }
} else if (Impact='Affects Multiple Users' or Impact='Affects Department') {
    if (Urgency='Low') {
        Set Priority to Low
    } else if (Urgency='Medium') {
        Set Priority to Medium
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
} else if (Impact='Affects Office') {
    if (Urgency='Low') {
        Set Priority to Medium
    } else if (Urgency='Medium') {
        Set Priority to High
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
} else if (Impact='Affects Company') {
    Set Priority to Critical
    if (Urgency='Low') {
        Set Priority to High
    } else if (Urgency='Medium') {
        Set Priority to High
    } else if (Urgency='High') {
        Set Priority to Critical
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
}
```

## Incident - All Creation Actions

### I: Conditional Creation Actions

```
if (Spawning SR ID!=NULL) {
    Update SR with conversion note
}
if (Problem CI Identified?='Yes' and Assigned Team=NULL and CI Responsible Team!=NULL) {
    Assign to CI Responsible team
} else {
    Assign to 1st level support team
}
```

### I: Notify Customer and Assignee

```
if (Status!='Closed') {
     Send Submitter creation acknowledgment email
     if (Assigned Person!=NULL) {
          Email Assigned Person of new Incident
     } else {
          Email Assigned Team of new Incident
     }
}
```

## Incident - Edit actions (API Enabled)

### I: Notification Actions for Incidents

```
if (Problem Closed? changed from: any value to: 'Yes' during record's last modification) {
     Email Incident Assignee of Problem Closing
}
if (Updated By!='$assigned_person' and Assigned Person changed from: any value to: any value during record's last
modification or Assigned Team changed from: any value to: any value during record's last modification) {
     if (Assigned Person!=NULL) {
          Email Assigned Person of new assignment
     } else {
          Email Assigned Team of new Incident
     }
     if (Assigned Team changed from: any value to: any value during record's last modification) {
          Increment Number of Assignees
     }
     if (Assigned Person changed from: any value to: any value during record's last modification) {
          Increment Number of Teams Assigned
     }
}
if (Status changed from: any value to: 'Workaround Provided' during record's last modification) {
     Email submitter and assignee of Workaround
}
```

### I: Conditional Assignment Actions

```
if (Assign to CI Responsible Team changed from: any value to: 'Yes' during record's last modification and CI Responsible
Team!=NULL) {
     Set Assigned Team to CI Responsible Team
}
```

## Incident Customer Update Actions

Run upon Edit, no API

Saved Search: Updated by Submitter

### I: Submitter Update Actions

```
if (Status='Closed' and I would like to reopen my Incident='Yes') {
        Change Status from Closed to Reopened
        Increment # of Reopens field
} else if (Status='Pending Customer') {
        Change Status to Update by Customer
}
if (Assigned Person!=NULL) {
        Email Assigned Person submitter update
} else {
        Email Assigned Team submitter update
}
```

## FORMATTING FOR ACTION BUTTONS

### Add Time

**Purpose:** Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the current Change Request record, then blanks out Time Spent and Time Description.

**On Click:** Save record, then execute action.

**Actions:**

#### I: Add Time

```
if (Time Spent=NULL) {
        Time spent must not be blank
} else {
        Convert to Time Entry
        Set Time Spent and Time Description to Blank
}
```

### Convert to Change Request

**Purpose:** Saves changes to the Incident and maps some of its fields into a new Change Request.

**On Click:** Save record, then execute action.

**Actions:**

#### C: Convert to Change Request

**Conversation:** Change Request

**Options:** Interactive, showing user the new record page

### Convert to Problem

**Purpose:** Save before and after conversion, prevent conversion if the problem ID already has a value.

**On Click:** Save record, execute action, save record again.

**Actions:**

### I: Convert to Problem if first conversion

```
if (Linked Problem ID!=NULL) {
     Only one linked Problem allowed
} else {
     Convert to Problem
}
```

## Convert to Service Request

**Purpose:** Silently convert to service request.

**On Click:** Execute action, then save record.

**Actions:**

### C: Convert to Service Request from Incident

**Conversion:** Service Request

**Options:** Silently with no confirmation

## Set Priority

**Purpose:** Automatically sets priority based on impact and urgency.

**On Click:** Execute action.

**Actions:**

### I: Set Priority

```
if (Impact='Affects Single User') {
    if (Urgency='Low') {
        Set Priority to Low
    } else if (Urgency='Medium') {
        Set Priority to Medium
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to High
    }
} else if (Impact='Affects Multiple Users' or Impact='Affects Department') {
    if (Urgency='Low') {
        Set Priority to Low
    } else if (Urgency='Medium') {
        Set Priority to Medium
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
} else if (Impact='Affects Office') {
    if (Urgency='Low') {
        Set Priority to Medium
    } else if (Urgency='Medium') {
        Set Priority to High
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
} else if (Impact='Affects Company') {
    Set Priority to Critical
    if (Urgency='Low') {
        Set Priority to High
    } else if (Urgency='Medium') {
        Set Priority to High
    } else if (Urgency='High') {
        Set Priority to Critical
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
}
```

## SAVED SEARCHES

Active Default searches are detailed below. The My Assigned search is the search used by the Inbox.

| Saved Search Name | Search Description |
|---|---|
| Closed Incidents | Finds all records where the Status is "Closed" |
| My Assigned | Assigned Person = global.name OR Assigned Team contains global.my_teams |
| My Incidents | Submitter Login='system' |

| Pending Customer Response | Status='Pending Customer' |
|---|---|
| Published | Publish to Knowledgebase='Yes' |
| Status is not Closed | Status!='Closed' |
| Updated by Submitter | Updated By='$customer_name' and Submitter Team~='customer' |

# INSURANCE CERTIFICATES TABLE

## FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|
| Amount of Coverage | Currency | Round to 2 decimal digits Display: 10 x 1; always show 2 decimal digits; $ |
| Attached Files | File with Versioning | Use to attach scanned/digital copies of the insurance certificate or other supporting documents. |
| Company Name, Main Contact, Main Contact Email, Main Contact Phone | Link to selected fields from other table:Companies Allow multiple values:No Allow entries not in source table:No | Link type: Strict Update: Automatically in the background Default: The Company matching the Contract Company field |
| Contract Company, Contract Amount, Contract Description, Contract End Date, Contract Start Date, Contract Title, Contract ID, Contract Status | Link to selected fields from other table:Contracts Allow multiple values:No Allow entries not in source table:No | Mapping: Strict Link Update: Automatically in the background |
| Expiration Date | Date | Used to keep track of the insurance certificate expiration date. Used in a time-based rule that controls the certificate status change. No Default; Required |
| Other Type | Short Text | Max Length: 70 Display: 40 x 1 Conditional Visibility: Type of Certificate == 'Other' Used to indicate the type of certificate if it does not match a choice on the drop- |

| | | down list. |
|---|---|---|
| Status | Choice | Choice List: Insurance Cert Statuses<br>Default: Valid |
| Type of Certificate | Choice | Choice List: Type of Certificate When Type == Other, a short text field becomes visible to fill a type not listed. |

## RULES AND ACTIONS

### TB: (Disabled) Demo Data Date Updates

**Purpose:** Keeps Demo Data updated with correct dates so reports work correctly.

**Runs:** Every month on the 1st at 2 AM

**Priority:** 1

**Saved Search:** Demo Data

**Action:**

#### U: Update Demo Dates

**Updated Fields:** Date Created, Expiration Date

**Updated Values:** $date_created + 1 MONTHS, $expiration_date + 1 MONTHS

### TB: Handle Insurance Certificate Expiration (Disabled by Default)

**Purpose:** Sets the status to Expired for contracts that were Active and past their expiration dates, and catches any other contracts past expiration and sets their status to Inactive.

**Runs:** Every 2 days at 2 AM

**Saved Search:** Valid Certificates Expiring Today

**Actions:**

#### I: Handle Certs Expiring Today

```
if (Contract Status='Active') {
     Set Status to Expired
     Email Contract Manager of expired certificate
} else {
     Set Status to Inactive
}
```

## TB: Notify Contract Manager Team 10 days in advance of expiration (Disabled by Default)

**Purpose:** Notify Contract Manager Team 10 days in advance of expiration date of an Insurance Certificate.

**Runs:** Every 2 days at 2 AM.

**Saved Search:** Cert is valid and expiring in 10 Days

**Actions:**

### I: Handle 10 Day Expiration Notices

```
if (Contract Status='Active') {
      Email Contract Manager of upcoming cert expiration
} else {
      Set Status to Inactive
}
```

## ITEMS TABLE

### FIELDS

| Field Label | Data Type | Comments |
|---|---|---|
| Active? | Choice: Active, Inactive | Default: Active |
| CI Class | Choice: Asset Class (see CI table) | Required |
| Item Name | Short Text 40/90 | Required, Unique, Summary field |
| Item Type | Choice: Hardware Peripheral Phone/PDA Software Furniture Services Other | Required |
| Require Supervisor Approval? | Choice: Yes/No | Default: No |
| Type of Item Service Categories | Link to selected fields in Service: Service Title, Types of Service | Filter by Saved Search: Show in Table = Purchase Request AND Status = Active |
| Unit Price | Currency: $ | Round, display 2 decimal digits Required, Default value: 0 |

### SAVED SEARCHES

**F:Active Items:**  Active? = Active

# ITEMS REQUESTED TABLE

## FIELDS

| Field Label | Data Type | Comments |
|---|---|---|
| Assign to User | Action Button | Text hyperlink: "Assign to Requestor"<br>Execute Action: Assign to User - Linked Update of CI<br>Then: Open Record for Edit |
| CI Class<br>Item Name<br>Type of Item<br>Unit Price<br>Require Supervisor Approval?<br>Service Categories | Link to selected fields in Item:<br>Item Name<br>Type of Item<br>Unit Price<br>CI Class<br>Require Supervisor Approval<br>Service Categories | Do not update<br>Filter by Saved Search: Active? = Active |
| CI Name<br>CI Description<br>Manufacturer<br>Model | Link to selected fields in CI:<br>CI Name<br>Description<br>Manufacturer, Model | Filter by:<br>Saved Search: F: In Stock CIs<br>AND Field Match:<br>CI Class = Type of Item |
| Available PR Asset Classes<br>Purchase Request ID<br>PO Number<br>Requestor Name<br>Requestor Login<br>Purchase Request Status<br>Type of Purchase | Link to selected fields in Purchase Request:<br>ID<br>PO Number<br>Requestor Name<br>Requestor Login<br>Status<br>Type of Purchase | Linked Purchase Request. If creating the Item Requested from a Purchase Request embedded table, these fields will auto-populate with that Purchase Request's information. |
| Quantity | Integer | Hidden by default to maintain link to CI.<br>Required, Default value: 1<br>Input box 3 chars wide, without a comma |
| Total Cost | Currency: $ | Hidden by default.<br>Round, display to 2 decimal digits |

## FORMATTING FOR ACTION BUTTONS

### Assign to User

**Purpose:** Update Configuration Item record with appropriate data.

**On Click:** Execute action, then save record and open for editing

**Action:**

#### L: Assign to User – linked update of CI

**Chain:** Item Requested/CI Name -> Configuration Item

**Linked Action:** U: Copy Req. Login to CI User Login, Update Status and Ownership

**U: Copy Req. Login to CI User Login, Update Status and Ownership**

**Updated Fields:** Ownership, Status, User Login

**Updated Values:** Assigned to an Internal User, Pending Install, "$item_requested.requester_login"

# LEADS TABLE

## FIELDS

*Required fields are denoted by an * next to the field label*

*(No fields on the Leads table are currently marked as required.)*

| Field label | Data Type | Admin notes |
|---|---|---|
| # of Employees | Integer | |
| Actions Taken | Multi-Choice | Choose From:<br><br>Sent Literature, Confirmed Available Budget, Identified Decision Makers, Qualified, Created Opportunity, Gave Webinar or Demo, Made Onsite Presentation, Followed Up After Demo, Sent Proposal, Negotiated Price, Closed Deal, Converted to Customer, Marked as Dead, Did Failure Follow-up |
| Address Notes | Text | |
| Amount of Sale | Currency | |
| Annual Revenue | Currency | |
| Assigned Sales Rep | Link to single field from other table:People<br>Allow multiple values:No<br>Allow entries not in source table:No | Links to the employee sales rep assigned |
| Assigned Team | Link to single field from other table:Teams<br>Allow multiple values:No<br>Allow entries not in source table:No | Defaults to Sales team<br><br>Links to the sales rep's team |
| Attached File(s) | File | |
| Billing Address 1 | Short Text | |
| Billing Address 2 | Short Text | |

| Billing City | Short Text | |
|---|---|---|
| Billing Country | Choice | Defaults to USA |
| Billing US State | Choice | |
| Billing Zip or Postal Code | Short Text | |
| Buying Stage | Choice | |
| Campaign Name, Campaign ID | Link to selected fields from other table:Campaigns Allow multiple values:No Allow entries not in source table:No | |
| Cell Phone | Telephone/Fax | Lead's cell phone |
| City | Short Text | |
| Company Name | Short Text | Lead Company Name |
| Country | Choice | Country of Origin for Company |
| Currency | Short Text | Defaults to US $ |
| Direct Phone | Telephone/Fax | For lead's direct phone line |
| Earliest Close Date | Date | |
| Email | Email | Lead's email address |
| Email Clicks from Outbound Emails | Related Table : Email Clicks | |
| Email Opt Out | Multi-Choice | Flag to allow opt-out from email lists |
| Fax | Telephone/Fax | |
| Find local hotels | Action button | |
| First Name | Short Text | First Name of Lead |
| Full Name | Compound | Lead's Name |
| How did you hear about us | Choice | Choose From: Webinar, Trade show, Friend referred me, Advertisement, Google, Other web search engine, Received email, Partner website, Marketing campaign, Purchased list, Partner referral, Customer referral, Employee referral, Other<br><br>For tracking advertising effectiveness and lead generation sources |
| Industry | Choice | Choose From: Aerospace & Defense, Agriculture, Automotive, Biotechnology & Drugs, Communications, Computer Hardware, Construction, Consumer Goods, |

| | | Education, Electronics, Financial Services, Food & Beverages, Government, Insurance, Leisure, Manufacturing, Medical & Health Services, Metals & Mining, Not for Profit, Printing and Publishing, Retail, Services, Software, Transportation & Travel, Utilities & Energy, Other |
|---|---|---|
| Intl State or Province | Short Text | |
| Key Requirements | Multi-Choice | Choose From: Fast deployment, Full reporting, Ease of use, Breadth of functionality, Customizability<br><br>Tracks lead priorities when looking at a product |
| Last Name | Short Text | Last Name of Lead |
| Latest Close Date | Date | For tracking upcoming sales volumes |
| Lead Created Via | Choice | Choose From: Self-registration, Added by Sales Staff<br><br>Defaults to: Self-registration<br><br>For tracking automated vs. manual lead creation |
| Map it | Action button | Opens URL:<br><br>"http://maps.google.com/maps?f=q&q=" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state |
| Negative Factors | Multi-Choice | Choose From: Price, Financing Terms, Features, Service, Guarantee, Demo, Documentation, Company Reputation, Personal Relationship, Other |
| Next Action | Choice | Choose From: Qualify, Send Literature, Confirm Available Budget, Identify Decision Makers, Create Opportunity, Schedule Presentation or Site Visit, Give Online Demo, Give Onsite Demo, Provide Proposal, Negotiate Price, Follow Up, Closed, Dead |
| Next Action Due | Date/Time | Reminder for the next action's logical due date |
| Opportunity Name | Short Text | For giving possible deals a unique name |
| Parent Company | Link to single field from other table:Companies<br>Allow multiple values:No<br>Allow entries not in source | For smaller subdivisions of larger companies, can search from current company records |

|  | table:Yes |  |
|---|---|---|
| Positive Factors | Multi-Choice | Multi-Choice: Price, Financing Terms, Features, Service, Guarantee, Demo, Documentation, Company Reputation, Personal Relationship, Other<br><br>For sales team tracking of what reasons a deal goes through |
| Probability of sale | Percentage | Likelihood of sale |
| Product | Choice | Choice: SupportWizard, Agiloft<br><br>For tracking companies that might have purchased your product |
| Public Company | Choice | Choice: Yes/No |
| Quotes | Related Table : Quotes |  |
| Rating | Choice | For tracking lead's interest level |
| Region | Choice | Choice: Western US, Central US, Eastern US, Canada, Europe, Asia, Africa, Other<br><br>Assists sales team in tracking company's time zone |
| Sale Comments | Append Only Text | For internal note taking on lead quality, status, and progress |
| State or Province | Short Text |  |
| Status | Choice |  |
| Street Address 1 | Short Text |  |
| Street Address 2 | Short Text |  |
| Telephone | Telephone/Fax | Lead's company main phone line |
| Title | Short Text |  |
| Type of Request | Choice | Choice: Download Program, Start Online Trial Program, Read Whitepapers, Newsletter Subscription, Join Beta Program, View Online Tutorial, View Online Demo, Sign up for Webcase,<br><br>Auto populates when lead record is created by guest actions |
| US State | Choice |  |
| Website URL | URL | For lead's company website |
| Zip or Postal Code | Short Text |  |

## RULES AND ACTIONS

### Lead Conversion

**Purpose:** Runs conversion of Lead to Company, Opportunity, and Contact records when Status of Lead record is set to "Converted."

**Runs:** Edit by Email, Web or API

**Search Filter:** Just Converted

**Priority:** 2

**Actions:**

#### C: Convert to Company, Opportunity, Contact

**Conversions:** People, Opportunity, Company

**Options:** Silently with no confirmation

### Campaign actions

**Purpose:** Automatically sets campaign name from name of lead.

**Runs:** Create by Email, Web, API

**Search Filter:** R: No campaign but user entry

**Priority:** 1

**Actions:**

#### U: Update Campaign from How did you hear

**Field Changed:** Campaign Name

**Value Changed to:** $lead_source

## FORMATTING FOR ACTION BUTTONS

### Find Local Hotels

**Purpose:** Searches Google Maps for hotels near the lead.

**On Click:** Opens URL: "http://maps.google.com/maps?f=q&q=hotels" + "+loc:+" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state

### Map It

**Purpose:** Searches Google Maps to find the location of the lead.

**On Click:** Open URL: "http://maps.google.com/maps?f=q&q=" + urlEncode($street_address_1) + ",+" + urlEncode($city) + ",+" + $us_state

## SAVED SEARCHES

The default searches provided for this table are detailed below.

| Saved Search Name | Search Description |
|---|---|
| Converted Leads | Finds Leads who have already been converted to Contacts, Companies, and Opportunities |
| Email Opt-out | Finds any Leads who have opted out from email contact |
| Just Converted | Finds all Leads where the status changed to "Converted" during the last modification |
| My Assigned | Finds Leads assigned to the logged-in user that have not been converted. |
| Unconverted Leads | Finds any Leads who haven't been converted to Contacts, Companies, and Opportunities |

# OPPORTUNITIES/DEALS TABLE

## FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments |
|---|---|---|
| *Opportunity Name | Short Text | For naming a specific opportunity in the event that there are multiple |
| *Product | Choice: Agiloft, SupportWizard | For tracking an Opportunity's product interests or a Deal's product of contract |
| Account Rep<br>Rep Phone<br>Rep Email<br>Rep Team | Link to fields from Employee table: Direct Phone, Email, Full Name, Primary Team | For information tracking the associated sales rep |
| Actions Taken | Search and import Multi-Choice:  Sent literature, Confirmed available budget, Identified decisions makers, Qualified, Created Opportunity, Gave Webinar or Demo, Made Onsite Presentation, Followed up after demo, Sent Proposal, Negotiated Price, Closed Deal, Converted to Customer, Marked as Dead, Did Failure Follow-Up | For tracking actions associated with sales opportunities |

| Additional Contacts (in Opportunities ) Contacts (in Deal subtable) | Linked data from Contacts table | Lists all records from the Contacts table associated with this Opportunity's linked Company |
|---|---|---|
| Value of Sale | Currency | The numerical value of the sale |
| Attached File(s) | File | For attaching related files |
| Buying Stage | Choice: Recognition of needs, Evaluation of options, Resolution of concerns | *Defaults to: Recognition of Needs* Stage in decision process |
| Company Billing Address 1 Billing Address 2 Billing City Billing Country Billing US State Billing Zip or Postal Code # of Employees Telephone, Fax Website URL State or Province Street Address 1 Street Address 2 Zip or Postal Code City USA State Country Currency Region Industry | Link to fields from Company table: # of Employees, Billing Address 1, Billing Address 2, Billing City, Billing Country, Billing US State, Billing Zip or Postal Code, City, Company Name, Country, Currency, Fax, Industry, Parent Company, Region, State or Province, Street Address 1, Street Address 2, Telephone, US State, Website URL, Zip or Postal Code | For tracking company information associated with the Opportunity or Deal |
| Campaign Name Campaign ID | Link to selected fields from other table: Campaign | |
| Earliest Close Date | Date | *Defaults to the record creation date* For estimating upcoming sales volumes |
| Key Requirements | Multi-Choice: Fast deployment, Full reporting, Ease of use, Breadth of functionality, Customizability | Tracks lead priorities when looking at a product |
| Latest Close Date | Date | For estimating upcoming sales volumes |

| Negative Factors | Multi-Choice: Price, Financing Terms, Features, Service, Guarantee, Demo, Documentation, Company Reputation, Personal Relationship, Other | For sales team tracking of what reasons a deal does not go through |
|---|---|---|
| Next Action | Choice: Qualify, Send literature, Confirm available budget, Identify decision makers, Create Opportunity, Schedule Presentation or site visit, Give Online Demo, Give Onsite Demo, Provide Proposal, Negotiate Price, Follow up, Closed, Dead | For tracking the next logical step in the sales process |
| Next Action Due | Date/Time | Reminder for the next action's logical due date |
| Positive Factors | Multi-Choice: Price, Financing Terms, Features, Service, Guarantee, Demo, Documentation, Company Reputation, Personal Relationship, Other | For sales team tracking of what reasons a deal goes through |
| Principal Contact Lead Source | Link to selected fields from Contact table: Full Name, Lead Source | For tracking the main person and lead generation associated with this sale |
| Probability of sale | Percentage | Likelihood of sale |
| Quotations | Linked data from another table: Quote | Filtered for Quotations related to this Opportunity/Deal |
| Rating | Choice: Active, Hot, Warm, Cold | For tracking opportunity interest level |
| Sale Comments | Append Only Text | For internal notes on lead quality, status, and progress |
| Tasks | Linked data from the Tasks table | For tracking Activities associated with closing a deal or maintaining a contract, such as demonstrations, onsite visits, and follow-up calls |
| Status | (In Opportunity) Choice: Open, Closed, Closed Won<br><br>(In Deal) Choice: Setup Needed, Setup Complete, Up for Renewal, Renewed, plus above | *Opportunity defaults to Open*<br><br>*Deal defaults to Setup Needed*<br><br>For tracking stage of each record |

| Fields found in the Deal Sub-table ONLY | | |
|---|---|---|
| Contract Notes | Text | For any associated notes |
| Contract Status | Choice: Valid, Expired, No Incidents Remaining, No Contract | *Defaults to: No Contract*<br><br>For tracking contract type |
| Expiration Date | Date | For annual contracts, tracks contract expiration date |
| Incidents Remaining | Integer | For tracking remaining incidents before a renewal |
| Original Start Date | Date | *Defaults to the record's creation date* |
| Renewal Number | Integer | The number contract renewals, for tracking contract longevity |
| Required Response Time | Elapsed Time | For tracking SLA response time requirements, such as response within 24 hours |
| Start Date | Date | *Defaults to the record's creation date* |
| Support Hours | Choice: 24/7, Standard Business Hours, Special Hours | Tracks the type of support chosen by a Deal/contract |
| Total Incidents | Integer | For tallying incidents for reporting on incident-based service Deals/contracts |
| Type of Support | Choice: None, Number of Incidents, Annual, Other | *Defaults to None*<br><br>Tracks the type of support purchased in a Deal |

## OPPORTUNITY SAVED SEARCHES

The default searches provided for this table are detailed below.

| Saved Search Name | Search Description |
|---|---|
| Assigned to My Team | Returns all records assigned to the logged-in user's team that are not closed |
| Converted to Deal | Shows all records where the Opportunity workflow state is set to Closed Won |
| My Assigned | Returns all records assigned to the logged-in user, where the workflow state is not Closed |
| Opportunities Only | Displays only records where the Type is set to "Opportunity" (excludes |

| Deals) | | | |
|---|---|---|---|

# PEOPLE TABLE

## FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments | In Table |
|---|---|---|---|
| *Login | Short Text | For contact's login to the system | User |
| *Password | Password | For the password associated with the above login | User |
| Actions Taken | Multi-Choice:  Sent literature, Confirmed available budget, Identified decisions makers, Qualified, Created Opportunity, Gave Webinar or Demo, Made Onsite Presentation, Followed up after demo, Sent Proposal, Negotiated Price, Closed Deal, Converted to Customer, Marked as Dead, Did Failure Follow-Up | For tracking actions associated with sales and marketing to a contact | User |
| All Document IDs Converted | Text | This field is used to hold all Document IDs entered and used to prevent the template from being converted again. Maximum size: 100,000; Display size: 90x5; Visible if Related to is Document | |
| Approval Source | Choice | Default: Document Management | |
| Assets | Related Table | Shows CI owned by a user | User |
| Buying Stage | Choice: Recognition of needs, Evaluation of options, Resolution of concerns | *Defaults to: Recognition of Needs*  Stage in decision process | User |
| Cell Phone | Telephone/Fax | For contact's cell number | User |
| Changes Request Submitted | Related Table | Show all records from the Change Request table submitted by user | User |

| Comments | Text | Additional information about the contact | User |
|---|---|---|---|
| Company Account Rep City Company Company Role Country Fax Region State or Province Street Address 1 Street Address 2 Company Phone USA State Zip or postal code | Link to fields from Company table: City, Company Name, Country, Company Role, Region, State or Province, Street Address 1, Street Address 2, Fax, Phone, US State, Zip or Postal Code | Tracks basic information related to the contact's company | User |
| Contact History | History | History of changes | User |
| Contact Owner | Link to single field from multiple tables Team: Team Name Employee: Full Name | Selects the team or individual responsible for the contact | User |
| Contact Table | Type of the SW object | | User |
| Direct Phone | Telephone/Fax | For the contact's direct phone line | User |
| DN | Short Text | This field is used by the system's LDAP integration, and should not be edited or deleted | User |
| Email | Email | Contact's email address | User |
| Email Clicks from Outbound Emails | Linked Data from Email Click table | Keep track of clicks for marketing purposes | User |
| Export to Outlook | Action Button | Opens URL: "/gui2/vCard;jsessionid=?id=" + urlEncode($id) | |
| First Name | Short Text | Contact's first name | User |
| Full Name | Compound | Combines the contact's first and last name fields | User |

| Groups | Link to single field from other table Group: Group Name | Lists the groups that the contact is a member of | User |
|---|---|---|---|
| ID | Auto-Increment | Automatically created by the system, this is the record's unique ID number | User |
| Key Requirements | Multi-Choice: Fast deployment, Full reporting, Ease of use, Breadth of functionality, Customizability | Tracks a contact's priorities when looking at a product | User |
| Last Document ID | Link to selected fields from other table: Documents Allow multiple values: No Allow entries not in source table: No | This field is used for the conversion to approvals. Visible if Related To is Documents | User |
| Last Document ID Text | Short Text | Used in the conversion to Approvals; Maximum size: 15; Display size: 15; Visible if Related To is Documents | User |
| Last Name | Short Text | Contact's last name | User |
| LDAP User | Choice: Yes, No | Flags if a contact was imported from an LDAP directory | User |
| Lead Created Via | Choice: Self-registration, Added by Sales Staff | *Defaults to: Self-registration*<br><br>For tracking automated vs. manual contact or lead creation | User |
| Lead Source | Choice: Webinar, Trade show, Friend Referred Me, Advertisement, Google, Other Web Search Engine, Received Email, Partner Website, Marketing Campaign, Purchased List, Partner Referral, Customer Referral, Employee referral, Other | Tracks a lead's source, imported into the Contact record during conversion | User |
| Next Action | Choice: Qualify, Send literature, Confirm available budget, Identify decision makers, Create Opportunity, Schedule Presentation or site visit, Give Online Demo, Give Onsite Demo, Provide Proposal, Negotiate Price, Follow up, Closed, Dead | For tracking the next logical step in the sales process | User |
| Next Action Due | Date/Time | Reminder for the next action's logical due date | User |
| Open Tasks | Linked data from another table | For tracking associated sales or marketing | User |

| | | activities that have not yet been completed | |
|---|---|---|---|
| Opportunities | Link to selected fields from other table Opportunity: Company, Opportunity Name | For tracking Opportunities connected with this Contact's company | User |
| Opt Out | Multi-Choice checkbox: Do not email me | Opt out of email | User |
| Primary Team | Link to single field from Team table: Team Name | Lists contact's primary team | User |
| Quotes | Related table | Lists Quotes for this contact | User |
| Rating | Choice: Active, Hot, Warm, Cold | For tracking contact's interest level | User |
| Reference as of | Date | Auto populated, set to the date that  Use as Reference was set to yes | User |
| Related To | Choice | Used in conversion process. Default: Documents | User |
| Rep Phone Rep Email Account Rep Sales Rep Team | Link to selected fields from other table Employee: Direct Phone, Email, Full Name, Primary Team | Links to the Employee responsible for sales and marketing to this Contact | User |
| Skype Login | Short Text | | |
| Status | Choice: New, In Progress, Qualified and Converted, Dead | *Defaults to New* Tracks lead status, mapped from Lead during conversion | User |
| Tasks History | Linked data from Tasks table | Tracks activities such as demos or onsite visits given to the contact | User |
| Teams | Link to single field from other table Team: Team Name | For tracking Teams that the contact belongs to | User |
| Title | Short Text | Contact's job title | User |
| Twitter Screen Name | Short Text | | |
| Twitter User ID | Integer | | |
| Type of Contact | Choice: Prospect, Customer, Partner, Reseller, Media, Vendor, Employee, Analyst, Investor, Other | *Defaults to  Customer* Tracks the type of Contact in more detail than | User |

| | | Employee/End User | |
|---|---|---|---|
| Updated By Updater Team | Link to selected fields from Contact table: Full Name | Automatically populated when the record is edited | User |
| Use as Reference | Choice: Yes, No | Flag a contact as a good business reference | User |
| User time zone | Choice | Specified time zone | User |
| Working Status | Choice | | User |
| Activity History | Link to Task Table | Displays all the Tasks the Employee has worked on | Employee |
| Department Name, Department ID, Department Manager | Link to selected fields from other table: Department | For tracking Employee's Department (outside of Team membership) | Employee |
| Email Pager | Email Pager | Email address for an emergency pager or similar | Employee |
| Employment Status | Choice: Active, Inactive | *Defaults to Active*<br><br>For tracking status | Employee |
| Ends Work at | Time | *Defaults to 17:00:00*<br><br>For tracking Employee work hours | Employee |
| Helpdesk Cases | Linked data from Service Request | Displays all the Service Requests the Employee submit | Employee |
| Hire Date | Date | Employee hire date | Employee |
| Home Phone | Telephone/Fax | For tracking Employee home (as opposed to work) contact number | Employee |
| Home Street Address Home City Home Zip Code | Short Text | For tracking Employee residence information | Employee |
| Manager Backup Manager Email Manager Name | Link to selected fields from other table Employee: Email, Full Name | Link to the Employee's manager | Employee |
| Picture | Image | Provide a picture | Employee |

| Starts Work at | Time | *Defaults to 08:30:00*<br><br>For tracking Employee work hours | Employee |
|---|---|---|---|
| Termination Date | Date | Employee end date | Employee |
| Time Zone | Choice: Pacific Standard Time, Central Standard Time, Eastern Standard Time, GMT, (GMT + or - 12) | *Defaults to Pacific Standard Time*<br><br>For tracking Employee location and conversion of working hours | Employee |
| Supported User | Choice: Yes, No | *Defaults to Yes*<br><br>For tracking is a contact is currently a supported customer | End User |

## RULES AND ACTIONS

### Edit: Approval Record Creation

**Purpose:** Performs conversion of Employee records to Approval records for documents approval and updates fields keeping track of Approvals that have been created.

**Runs:** Edit by Web or API

**Search Filter:** R: Last Doc ID Text Changed and All Docs does not contain Last Doc ID

**Priority:** 1

**Actions:**

#### C: Convert Employee to New Approval

**Conversion:** Approval

**Options:** Silently with no confirmation

#### U: Add to All Document IDs Converted

**Fields Changed:** All Document IDs Converted

**Value Changed to:** $last_document_id_text+","

## FORMATTING FOR ACTION BUTTONS

### Export to Outlook

**Purpose:** Downloads the user's information to vcf format which will

Allow it to be saved and opened in MS Outlook

**On Click:** Open URL: "/gui2/vCard;jsessionid=?id=" + urlEncode($id)

# Problems Table

## Fields

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type/ Length/Default | Comments |
|---|---|---|
| Add Time | Action Button | Converts Time Spent and Time Description to a Time Entry record linked to this Problem. |
| Add to Knowledgebase? | Choice: Yes/No | Marks record as knowledge (FAQ)<br>Radio buttons 2 per row<br>Default: no |
| Additional Notes | Append-only | |
| All Time Entries | Link to data from Related Table: Time Entry | Filtered for Time Entries relating to this Problem |
| All Time Spent | Calculation on Multiple Linked Records | Sums the All Time Entries linked data. |
| Assigned Person | Link to Contact table: Full Name | Link to Contact |
| Assigned Solution Team | Link to Team | Default: Same as Diagnosis team<br>This is the team which tries to find a solution that will solve the root cause of the known error. This team is typically built using staff from the Diagnosis Team. |
| Assigned Team<br>Team Leader | Link to Team | This is the team which searches for the root cause of a problem and is the Problem Agent for the record.<br>Default: 1st Level Support Team |
| Attached Files | File | |
| Change Summary<br>Change Request ID | Link to Change Request table: Change Request ID, Change Category | View: Lookup Change ID, Summary, Category |
| Configuration Item<br>CI User Name<br>CI Class<br>CI Tag<br>CI Type<br>CI Serial Number<br>CI Status | Link to Configuration Item table: CI Name | Link to CI |
| Convert to Change Request | Action Button | Text hyperlink: "Save Changes and Copy to Change Request"<br>Execute Action:<br>```if (Change Request ID!='NULL') {`<br>`    Cannot have more than one linked CR`<br>`} else {`<br>`    Convert to Change Request`<br>`}```<br>Run Interactively, showing user results. |

|  |  | Then: Close record.<br>Input instruction: "Click this button if you want to save your changes to this Problem and map some of its fields into a new CR. If you want to link to an existing CR, use the lookups in the CR  fields below to locate the correct CR." |
|---|---|---|
| Converted from Incident | Choice: Yes/No | Default: No. |
| Copy Solution to Incident and Close it | Action Button | Text hyperlink: " Update Incidents with Solution"<br>Copies the contents of the Resolution field in the Problem record to the Resolution fields of linked Incidents |
| Copy Workaround to Incident | Action Button | Text hyperlink: " Update Incidents with Workaround"<br>Copies the contents of the Workaround field in the Problem record to the Resolution fields of linked Incidents. Sets the "Workaround Provided?" field in the Problem record to Yes. |
| CR Closed? | Choice: Yes/No | Flag for notification when linked CR closes. Not shown to users.<br>Default: No |
| Date Resolved | Date/Time | Default: Date Status changed to Resolved |
| Date to Pending Change | Date/Time | Default: Date Status changed to Pending Change |
| Description | Text 5000/90x6 |  |
| Diagnosis Finish Time | Date/Time | To track diagnosis time of Diagnosis team. Set by the Start Diagnosis Clock action button. |
| Diagnosis Start Time | Date/Time | To track diagnosis time of Diagnosis team. Set by the End Diagnosis Clock action button. |
| Elapsed Time from Pending Change to Resolved | Elapsed Time | Measure of the elapsed time between Date to Pending Change and Date Resolved. |
| Elapsed Hours from Diagnosed to Resolved | Elapsed Time | Measure of the elapsed time between Diagnosis Finish Time and Date Resolved. |
| End Diagnosis Clock | Action Button | Text hyperlink: "End Clock"<br>Action: Set Diagnosis Finish Time to Now |
| Incidents | Related table: Incidents | Display using: Embedded View<br>Shows Incident records linked to this Problem |
| Known Error | Choice: Yes/No | Default: No |

| | | Marks this Problem as a Known Error. |
|---|---|---|
| Priority | Choice – Low, Medium, High, Critical | Required<br>Default: Medium |
| Problem Priority | Choice: Standard (0), Minor (1), Significant (2), Critical (3) | Default: Standard (0)<br>Pop-up text: A measure of overall impact and risk. Reflects level of authorization needed. |
| Problem CI Identified? | Choice: Yes/No | Default: No |
| Resolution | 5000/90x6 | Solution field |
| Risk Analysis | Text  2000/90x4 | For staff to describe the risks associated with this problem and its potential resolutions. |
| Root Cause | Text 2000/90x3 | Popup text: Describe the root cause of the Problem here after successful diagnosis. |
| Service Involved | Choice list Incident Type of Problem:<br>Access Problem, Email, Network/Internet Connectivity, Custom Business Application, Desktop Application, Desktop Hardware, Other Hardware or Peripheral Phone/Cell/PDA Printer | Required. |
| Set Solution Finish Time | Action Button | Text hyperlink: Stop Clock<br>Sets Solution Finish Time to Now |
| Set Solution Start Time | Action Button | Text hyperlink: Start Clock<br>Sets Solution Start Time to Now |
| Solution Finish Time | Date/Time | To track "solution building" time |
| Solution Start Time | Date/Time | To track "solution building" time of Solution team |
| Solution Files | File | For files related to problem's resolution |
| Spawning Incident ID | Link to selected fields in Incident: ID | For conversion linking, will hold the ID of the Incident record that was converted to this Problem |
| Start Diagnosis Clock | Action Button | Text hyperlink: "Start Clock"<br>Action: Set Diagnosis Start Time to Now |
| Status | wfstate | Status field<br>See **Workflow Diagram** |
| Summary | Short Text 200/90 | |
| Time Description | Text | Holds description of a Time Entry |
| Time Spent | Time | Holds time spent for a Time Entry |
| Total Time to Resolve (Hours) | Elapsed Time | Measure the elapsed time between Date Created and Date Resolved<br>Only show if: Status = Resolved |
| Workaround | Text 10000/90x6 | Description of work-around, if any |

| Workaround Provided? | Choice: Yes/No | Radio buttons, Default: No<br>Indicates that the Problem has a workaround supplied in the Workaround field |
|---|---|---|
| Working Diagnosis Hours | Elapsed Time | Measure of the elapsed time between Diagnosis Start Time and Diagnosis Finish Time |
| Working Solution Hours | Elapsed Time | Measure of the elapsed time between Solution Start Time and Solution Finish Time |

## SAVED SEARCHES

Default searches provided are detailed below.  The My Assigned search is used by the Inbox.

| Saved Search Name | Search Description |
|---|---|
| My Assigned | (Assigned Person = global.name<br><br>OR Assigned Team is contained in global.my_teams)<br><br>AND Status < Deferred |
| Known Errors | Known Error = Yes |

## RULES AND ACTIONS

### Rule: Problem - Creation actions

#### I: Conditional Creation Actions

```
if (Spawning Incident ID!=NULL) {
        Update "Additional Info" in Spawning Incident
}
```

### Rule: Problem - Edit actions (API Enabled)

#### I: Notification actions

```
if (Status changed from: any value to: 'Resolved' during record's last modification) {
        if (Spawning Incident ID!=NULL) {
                Set "Problem Closed?" in Incident to Yes
        }
}
if (CR Closed? changed from: any value to: 'Yes' during record's last modification) {
        Email Problem Assignee of CR Closing
}
```

## FORMATTING FOR ACTION BUTTONS

## Add Time

**Purpose:** Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the Service Request, then blanks the fields.

**On Click:** Save record, then execute action.

**Actions:**

### I: Add Time

```
if (Time Spent=NULL) {
       Time spent must not be blank
}
Convert to Time Entry
Clear Time fields
```

## Convert to Change Request

**Purpose:** Validates the problem has no change request associated with it, then converts to a change request.

**On Click:** Execute action, then save record.

**Action:**

### I: Convert to CR (if first conversion)

```
if (Change Request ID!=NULL) {
       Cannot have more than one linked CR
} else {
       Convert to Change Request
}
```

## Copy Solution to Incident and Close It

**Purpose:** Validates a resolution exists, then copies the resolution from the problem to a linked incident.

**On Click:** Execute action.

**Action:**

### I: Copy Solution field to related Incidents

```
if (Resolution=NULL) {
       solution must have content
} else {
       Copy Solution from Problem to Incident
}
```

## Copy Workaround to Incident

**Purpose:** Copies the contents of the Workaround field in the Problem record to the Resolution fields of linked Incidents. Sets the "Workaround Provided?" field in the Problem record to Yes.

**On Click:** Execute actions.

**Actions:**

### I: Update Linked Incident with Workaround

```
if (Workaround=NULL) {
       you must have content in workaround field
} else {
       Copy Workaround from Problem to Incident
}
```

### U: Set Workaround Provided to yes

**Updated Field:** Workaround Provided

**Updated Value:** Yes

## End Diagnosis Clock

**Purpose:** Sets Diagnosis Finish Time to "now."

**On Click:** Execute action.

**Action:**

### U: Set Diagnosis Finish Time to Now

**Updated Field:** Diagnosis Finish Time

**Updated Value:** NOW()

## Set Solution Finish Time

**Purpose:** Sets Solution Finish Time to NOW().

**On Click:** Execute action.

**Action:**

### U: Set Solution Finish Time to Now

**Updated Field:** Solution Finish Time

**Updated Value:** NOW()

## Set Solution Start Time

**Purpose:** Sets Solution Start Time to Now

**On Click:** Set Solution Start Time to Now

**Action:**

U: Set Solution Start Time to Now

**Updated Field:** Solution Start Time

**Updated Value:** NOW()

## Start Diagnosis Clock

**Purpose:** Sets Diagnosis Start Time to NOW().

**On Click:** Execute action.

**Action:**

U: Set Diagnosis Start Time to Now

**Updated Field:** Diagnosis Start Time

**Updated Value:** NOW()

# PROJECTS TABLE

## FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type | Comments |
|---|---|---|
| *Project Description | Short Text | Describe the project in detail |
| *Project Name | Short Text | Gives the project title or identifying name |
| *Type of Project | Choice: New Customer Implementation, Customer Upgrade, Additional Services for Existing Customer | Tracks the type of project |
| Add Time | Action Button | Commits time added to the Project as a Time Entry record linked to the Project |
| Attached Contract | File | For attaching contract documents and related files |
| Billing Notes | Append Only Text | For any notes related to billing or invoicing |
| Client Business Contact Business Contact Phone Business Contact Email | Link to selected fields from Contact table: Direct Phone, Email, Full Name | For tracking information of the client's business contact |

| | | |
|---|---|---|
| Client Technical Contact Title Company Cell Phone Direct Phone Email | Link to selected fields from Contact table: Cell Phone, Company Name, Direct Phone, Email, Full Name, Title | For tracking information of the client's technical contact |
| Contract Approved by Customer | Choice: Yes, No | Flags if a Project has been approved by the customer |
| Created By Creator Login Creator Team | Link to selected fields from Contact table: Full Name, Login, Primary Team | For tracking record creation info |
| Date Completed | Date/Time | Date the Status changed to Completed or Cancelled, do not update (so user can override). |
| Date Contract Approved | Date/Time | For tracking the date the contract was approved |
| Date Created | Date/Time | Automatically generated during creation |
| Date Updated | Date/Time | Automatically populated when record is modified |
| Default Tasks | Link to single field from other table: Task Templates | Lists the set of tasks that will automatically be generated because of the Project's type. |
| Developer Phone Developer Email Developer Name | Link to selected fields from Employee table: Direct Phone, Email, Full Name | Links to the employees working on this project |
| Estimated Hours | Integer | For cost estimation |
| Hours Spent | Linked data from another table | Tracks the number of Hours records associated with this Project |
| ID | Auto-Increment | Auto-incremented by the system; this is the record's ID number |
| Internal Project Notes | Append Only Text | Text box to keep notes in for internal users. |
| Next Milestone Date | Date | For tracking the next agreed upon milestone date |
| Number of Open Tasks | Calculation on Multiple Linked Records: Tasks | Counts non-closed Task records linked to this Project. Used by automation to set status of Project. |

| Number of Tasks | Calculation on Multiple Linked Records: Tasks | Counts Task records linked to this Project. Used by automation to set status of Project. |
|---|---|---|
| PM Phone, PM Email, Project Manager, PM ID, PM Login, PM Manager Name, PM Primary Team | Link to selected fields from Employee table: Direct Phone, Email, Full Name | For tracking internal Project Manager's contact information |
| Status | Choice: Awaiting Customer Feedback, Completed, Revisions in Progress, Ready for Testing, Work in Progress, Pending | *Defaults to Pending*<br><br>Tracks workflow state of the Project |
| Target Completion Date | Date | For the agreed upon completion date |
| Tasks | Related Table: Tasks | Shows Tasks linked to this Project. |
| Time Authorizations (POs) | Linked data from another table | Links to PO records authorizing work hours |
| Total Hours Spent | Calculation on Multiple Linked Records | Calculates the number of Hour records associated with this project |
| Total Invoiced Hours | Calculation on Multiple Linked Records | Calculates the hours on invoiced Hour records |
| Total Number of Hours Authorized | Calculation on Multiple Linked Records | Calculates the total authorized hours across all linked POs |
| Total Uninvoiced Hours | Calculation on Multiple Linked Records | Calculates the hours on uninvoiced Hour records |
| Total Value of Hours Authorized | Calculation on Multiple Linked Records | Calculates the total value of hours across all linked POs |
| Updated By<br>Updater Login<br>Updater Team | Link to selected fields from Contact table: Full Name, Login, Primary Team | Tracks name and login information of the last person to edit the record |

## RULES AND ACTIONS

### Edit by Web: All edit actions: trigger tasks, cancel tasks when cancelled, etc.

**Purpose:** Validates, triggers, and cancels tasks.

**Runs:** On Edit by Web.

**Search Filter:** None.

**Priority:** 2

**Actions:**

### I: All Edit Validations

```
if (Status changed from: any value to: any of 'Completed','Tasks Completed' last user's modification and Number of Open
Tasks>=1) {
    cannot complete while open tasks
}
```

### I: All Edit Actions

```
if (Default Tasks!=NULL and Status changed from: any value to: 'Work In Progress' last user's modification) {
    Trigger Default Tasks
} else if (Status changed from: any value to: 'Cancelled' last user's modification) {
    Cancel Tasks
    Email Project CCs about Cancelled Project
}
```

## TB: notify Project Manager if hours exceed authorized hours

**Puprose:** Checks on authorized hours and hours worked and sends notification if authorized hour are exceeded.

**Runs:** Every day.

**Search Filter:** R: total hours for client project exceeds authorized hours

**Priority:** 1

**Actions:**

### U: Set Status to Assigned

**Fields Updated:** Status

**Updated To:** Assigned

### E: Email PM project has exceeded hours

**Email Template:** Project $formula($project_name) has exceeded authorized hours

## Edit by Web or API: Handle completion or addition of tasks and status changes

**Purpose:** Opens or completes projects depending on number and state of tasks linked to project.

**Runs:** Edits by Web or API.

**Search Filter:** R: Has Tasks and the number of open tasks changed

**Priority:** 1

**Actions:**

### I: Handle completion or adding tasks

```
if (Number of Open Tasks=0 and Status='Work In Progress') {
     Set Status To Tasks Completed
     Email PM, Project CCs All Tasks Completed
} else if (Number of Open Tasks changed from: =0 to: > 0 during record's last modification and Status='Tasks Completed')
{
     Set Status to Work in Progress
}
```

## Create: All Project creation actions – trigger tasks

**Purpose:** Notify Project Manager and CCs of new project, and begin default tasks.

**Runs:** Creation by Email, Web, or API:

**Search Filter:** None.

**Priority:** 1

**Actions:**

### I: All project create actions

```
if (Creator Login!='$pm_login') {
     Email PM of new project assignment
}
if (Project CCs!=NULL) {
     Email project ccs of new project
}
if (Status='Work In Progress') {
     Trigger Default Tasks
}
```

## SAVED SEARCHES

The default searches provided for this table are detailed below.

| Saved Search Name | Search Description |
|---|---|
| Active Projects that have exceeded Authorized Hours | Finds all Projects in "Pending" status where the total hours are greater than the number of hours authorized |
| Company name match | Finds all records where the project's company matches that of the logged-in user |
| All My Projects | Finds all projects for which the user's login matches that of the Project Manager |

## PROJECT TYPES TABLE

### FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|

| Has Task Templates | Choice | Default is No, once a task template is created, this field is changed to Yes by a rule. |
|---|---|---|
| Number of Task Templates | Calculation on Multiple Linked Records : Task Templates | Filtered to status is Active |
| Project Category | Choice | choices: Internal, Client-Related, required |
| Project Type | Short Text | Maximum: 95, display: 60, summary field, required |
| Status | Choice | Default: Active, other choice: Inactive |
| Task Templates | Related Table : Task Templates | This table shows the Task Templates that are configured for this Project Type |

## RULES AND ACTIONS

### Create/Edit by Web or API: Update Has Task Templates

**Purpose:** Checks to see if Task Templates exist in this Project Type and sets the flag field Has Task Templates to Yes if so.

**Runs:** Create/Edit including rules by Web or API

**Search Filter:** R: Has task templates is no and number of templates is 1 or more

**Priority:** 10

**Actions:**

#### U: Set Has Task Templates to Yes

**Field Changed:** Has Task Templates

**Value Changed to:** Yes

## PURCHASE ORDERS TABLE

### FIELDS

*Required fields are denoted by an * next to the field label*

*(No fields on the PO table are currently marked as required.)*

| Field Label | Data Type | Comments |
|---|---|---|
| Attached File | File | For attaching related files such as invoices or estimates |
| Authorized/ Purchased By | Link to selected fields from Contact table: Full Name | Select a Contact who has approved this work. |
| Calculate Value | Action button | Multiplies the Hours by the Rate, and puts the result in the field "Total Value" |
| Company | Link to selected fields from Project table: | Tracks the Project and Company that |

| Project Name | Company, Project Name | the PO relates to |
|---|---|---|
| Date Authorized | Date | Fill in the date this PO was authorized or approved |
| Hours | Floating Point | *Defaults to 0* <br><br> Tracks the number of hours to be billed on this invoice |
| Linked Company | Link to selected fields from other table: Company | Company PO comes from |
| PO is for | Short Text | Short title for work authorized under this PO |
| PO Number | Short Text | For adding in an external PO number |
| Rate | Currency | Defaults to $200 <br><br> For tracking billed rate |
| Status | Choice: Paid, Fully Invoiced, Progress Invoiced, In Progress | *Defaults to In Progress* <br><br> Workflow state, tracks status of PO |
| Total Value | Currency | The PO total is calculated into this field |
| Work Description | Text | Describe the work to be done in this PO |

## FORMATTING FOR ACTION BUTTONS

### Calculate Value

**Purpose:** Multiplies the Hours by the Rate, and puts the result in the field "Total Value".

**On Click:** Execute action.

**Action:**

> ### U: Total Value = Hours * Rate
>
> **Updated Field:** Total Value
>
> **Updated Value:** $hours * $rate

## PURCHASE REQUESTS TABLE

### FIELDS

| Field Label | Data Type | Comments |
|---|---|---|
| Account Number | Short Text 30/50 | Required |
| Additional Notes | Append Only Text 90x7 | |

| All Items Received? | Choice: Yes/No | Default: No |
|---|---|---|
| Approver Phone<br>Approver Email<br>Approved/Rejected By | Link to selected fields in User:<br>Direct Phone<br>Email<br>Full Name | Default: User who changed Procurement Group Approved to any value |
| Assigned To | Link to selected fields in Employee: Full Name | |
| Business Justification | Text 5000, 90x6 | Required |
| Category | Choice:<br>Hardware<br>Peripheral<br>Phone/PDA<br>Software<br>Furniture<br>Services<br>Other | Hidden by default, replaced by Type of Purchase in Service linked set |
| Date Approved | Date/Time | Date Status changed to Approved by Procurement |
| Date Ordered | Date/Time | Date Status changed to Ordered |
| Date Received | Date/Time | Date Status changed to Received |
| Internal Notes | Append Only Text 90x7 | Only visible to internal staff |
| Items | Related Table: Items Requested | Used for requestor to select desired items |
| PO Number | Short Text 40/30 | |
| Priority | Choice:<br>Low<br>Medium<br>High Critical | Required |
| Procurement Group Approved | Choice: Yes/No | Visible to Procurement Group, rule sets Status |
| Requestor Name<br>Requestor Email<br>Req. Department<br>Req. Phone<br>Req. Login<br>Manager Name | Linked to selected fields in Employee | Identifies Requestor<br><br>Default: User who creates the Purchase Request |
| Service Action Required? | Choice: Yes/No | Hidden by default |
| Additional Fields to Show in SR<br><br>Available Service Asset Classes<br><br>CC Manager on Request<br>Description of Service | Link to Service table | Filter by Saved Search: Show in Table = Purchase Request AND Status = Active<br><br>AND Field Match:<br>Service Categories contains Service Category |

| Team Priority over CI Service Responsible Team Type of Purchase Service Categories SLA Hours - Critical Priority SLA Hours - Standard Priorities Special Instructions | | Required Display using: Embedded View All fields except Service are view only |
|---|---|---|
| Service Category | Choice: Type of Service (see Service table) | Dependent choices based on all statuses: Facilities Purchase Request, Software Purchase Request, Hardware Purchase Request |
| Status | Workflow State | See Workflow Diagram |
| Supervisor Approved | Choice: Yes/No | Supervisor sets this field when approving/rejecting, rule then sets status |
| Supervisor Approver Email Supervisor Approved/Rejected By | Link to selected fields in User: Email, Full Name | Default: User who changed the Supervisor Approved field to any value |
| Team Leader Assigned Team | Link to selected fields in Team | |
| Total Item Cost | Calculation on Multiple Linked Records | Sum of Unit Price of Requested Items |

## RULES AND ACTIONS

### Rule: Purchase Request - New Request Actions

#### Condition and Schedule

When a Purchase Request is Created by Web, Priority 1

#### I: Validation for New Requests

```
if (Supervisor Approved='Yes' and Created By='$approved_rejected_by0') {
     Only Supervisor Can Approve
}
```

#### I: New Purchase Request Actions

```
Email Requestor about new Purchase Request
Email Purchasing Team aboue new Purchase Request
Email Supervisor that PR requires approval
if (Supervisor Approved='Yes') {
        Set Status to "Supervisor Approved"
}
```

## Rule: Purchase Request - Edit Validation Actions

### I: Actions Based on Edit Validations

```
if (Status='Approved by Procurement' and (Supervisor Approved=NULL or Procurement Group Approved=NULL)) {
        Validate Approved Status Change
}
if (Status changed from: any value to: 'Rejected by Procurement' last user's modification and Additional Notes not changed
last user's modification) {
        Validate Additional Notes exist for Not Approved request
}
```

## Rule: Purchase Request - All Staff Edits

### I: Staff Edit Actions

```
if (All Items Received?='Yes' and Status!='Received') {
        Update Status to Received
}
if (Status!='Supervisor Rejected' and Supervisor Approved changed from: any value to: 'No' last user's modification) {
        Set Status to Supervisor Rejected
}
if (Status changed from: any value to: 'Rejected by Procurement' last user's modification) {
        Email Requestor about Not Approved Request
}
if (Supervisor Approved changed from: any value to: 'Yes' last user's modification and Status='Pending Approval') {
        Set Status to Supervisor Approved
}
```

## Rule: Purchase Request - All End User Edits

### I: Actions Based on End User Edits

```
if (Additional Notes changed from: any value to: any value last user's modification and Requester Login='$updater_login') {
        Email Purchasing Team about updated Purchase Request
}
```

# SERVICES TABLE

## FIELDS

| Field Label | Date Type | Comments |
|-------------|-----------|----------|
|             |           |          |

| Field Label | Date Type | Comments |
|---|---|---|
| Show in Table | Choice: Service Request, Change Request, Purchase Request | This is a parent field for several other fields in the Services table, defining which fields and choices are shown based on the table in which the service will be used. |
| Additional Fields to Show in SR | Multi-Choice | Visible only if Show in Table=Service Request.<br><br>This multi-choice field lists all the special fields or sets of fields in the Service Request table that should only be visible for specific types of service.  Any fields or sets of fields selected here will be made visible in the Service Request for this particular Service.  This field is pulled into the SR along with the Service and the relevant fields are dependent for their visibility on the value in this linked field.<br><br>Note: we have added some extra values to this field so that if you are setting up a new service and realize you need a new field or fields to show for that service, you can choose one of these "extra" field values, save your service, and then later edit the choice for this field to give it the proper name.  To add field names to the list or edit those that are there, simply edit this field and click on Edit Choice List on the General tab. |

| Field Label | Date Type | Comments |
|---|---|---|
| Available Asset Classes | Multi-Choice | Choose From: Application Server, Bar Code Scanner, Camera, Cell Phone, Cluster, Database Application, Database Instance, Desktop Application, Desktop Computer, Desktop Phone, Directory Server, Disk Drive, Email Server, Fax Machine, File Server, Firewall, FTP Server, Google Search Appliance, IBM Mainframe, Individual Printer, Infrastructure Software, IP Conference Phone, IP Device, IP Network, IP Router, IP Switch, Java Server, Keyboard, KVM Switch, Label Maker, Laptop Computer, Load Balancer, Modem, Monitor, Monitoring Device, Mouse, Network Adapter, Network Appliance, Network Printer, Network Switch, NFS File System, Operating System, Other Application, Other Hardware, PDA enabled Cell Phone, Peripheral – Other, Rack, Scanner, SMB File System, Storage Array, Storage Device, Storage Management Server, Storage Switch, Tape Library, UPS, Video Conferencing, Video Card, Virtual Machine, Virtual Private Network, Voice Gateway, VPM Appliance, WAN Interface, Web Application, Web Server Application, Web Service, Web Site, Wi-Fi___33, Windows Service, Windows Workstation |
| Service | Short Text, maximum 90 | Required, must be unique, summary field for table. The name of the service – it will be shown on a drop-down list in the Request |
| Description of Service | Text, max 5000 | This description will be shown in the Request when the service is selected. |
| Special Instructions | Text, max 5000 | Format: HTML Text This field may be used to provide special instructions to the end user telling her what special information is needed for a request for this service.  It may be formatted with color or other HTML formatting to make it stand out for the user.  It will be displayed in the new request form as soon as the user chooses the service. |

| Field Label | Date Type | Comments |
|---|---|---|
| Primary Service Category | Choice | Used to set visibility in submission of Service Request. |
| Service Categories | Multi-Choice: | Before choosing a service, a user will generally pick a Service Category from a drop-down list, and the list of services is then filtered to those that contain the Service Category that was selected in this field. Because some services might cross categories, we made this a multi-choice field so you can assign a service to more than one possible category.<br><br>The choices in this field are dependent on the values in the Show in Table field.  If you add new service categories, you will need to map the dependencies to make them show up for the correct value of that field. This is done on the Options tab of the field wizard. You can see and edit the current mappings at the bottom of the screen.  See services.xls for a list of default categories and mappings (you can modify this worksheet and do an update import to create the services and categories you want). |
| Responsible Team | Link to Teams | This is the team that should be assigned by default to any requests for this service. |
| Give Responsible Team priority over CI Team | Choice: Yes/No | Required.  Default Value: No.<br><br>This field affects the default assignment to a team for new service requests.  By default the logic works like this: if the Service request involves a specific configuration item, then the request is assigned to the responsible team for that CI.  If no CI is defined, then it is assigned to the Responsible team for the Service.<br><br>But for some services, the Service responsible team should take priority over any CI Team, and if that is the case, then you should set this field to Yes for that service.<br><br>Note that if a staff person assigns the new item to a specific team during record creation, his selection will not be overridden. |

| Field Label | Date Type | Comments |
|---|---|---|
| CC Manager on request | Choice: Yes/No | Visible if Show in Table is Service Request or Purchase Request.  Default value: No

This field is used to indicate that a submitter's manager should be automatically cc'd upon receipt of a request for this service.

For instance, you may not require manager approval before a user requests an application upgrade, but you may want the user's manager to know about it and to be able to reject or limit the request as appropriate.  Putting a Yes in this field will cause the manager to be cc'd by the system when the request is received so she can intervene if necessary. |
| SLA Hours - Critical Priority | Floating Point | Default: 2 hours

This field may be used to define the SLA due date for this service if the priority is set to Critical.  This field will be pulled into the related Request and used to set an SLA Due Date that can then trigger escalations and automate reporting on whether SLA targets are being met. |
| SLA Hours - Standard Priorities | Floating Point | Default: 24 hours

This field may be used to define the SLA due date for this service if the priority is set to anything except Critical.  This field will be pulled into the related Request and used to set an SLA Due Date that can then trigger escalations and automate reporting on whether SLA targets are being met.

Naturally if you want a different SLA Due date for each Priority value, you may create additional fields, one for each Priority value. |

| Field Label | Date Type | Comments |
|---|---|---|
| Number of Approvals Needed | choice: 1, 2, 3, 4, 5 | Visible only if Show in Tables=Change Request. Default: 1 <br><br> This field is used to indicate how many different approvals will be needed for change requests for this service.  It is pulled into the change request form and the value in this field determines how many approval fields are shown in the change request form. <br><br> It is editable in the change request table, so a person with permission access to edit the field there can choose a different number if needed. |
| Status | Workflow | Default:  Planned <br><br> The workflow states are shown in a diagram below.  To change the status choices and/or transitions, go to Setup/Workflow/ and choose the Services table and click Edit. |
| Types of CI | Multi-Choice | Flag field for other automation. |

## SAVED SEARCHES

| Saved Search Name | Search Description |
|---|---|
| Active Service Request Services sorted by category | Show in Table='Service Request' and Status='Active' |
| Inactive Services | Status='Inactive' |
| Planned Services | Status='Planned' |
| Purchase Request Services | Show in Table='Purchase Request' |
| Purchase Request Services and Status is Active | Show in Table='Purchase Request' and Status='Active' |
| Retired Services | Status='Retired' |
| Service Catalog | Status='Active' |
| Service Pipeline | Status='Planned' |

## SERVICE REQUESTS TABLE

### FIELDS

| Field Label | Data Type/Size | Comments |
|---|---|---|
| Add Time | Action Button | Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the Service Request, then blanks the fields. |
| Add to Knowledgebase? | Choice: Yes, No | Visible if Status=Closed or Reopened |
| Additional Fields to Show CC Manager Description of Service Give Service Team Priority over CI Team Service Responsible Team Service Service Categories SLA - Critical Priority Hours to Complete SLA Hours to complete Special Instructions | Link to Services | This linked field set provides the selection of the service category and service, and pulls in additional fields from the service table that define resolution times, SLA, additional fields that should be shown, and so on. |
| Additional Information | Append Only Text, 10000 | For communicating with customer and allowing the customer to provide updates |
| All Time Entries | Related Table: Time Entries | Shows all time entered for this SR |
| All Time Spent | calc on Related Table | Totals number of hours in related table of All Time Entries |
| Application, Application Responsible Team, Application Version | link to Configuration Item | Visible if Additional Fields to show includes Software Application Version or Software Application CI Fields. Filtered to CI's whose Class is Software Application |
| Assigned Person, Assignee Manager | Link to Employee | filtered to members of assigned team |
| Assigned Team, Team Pager, Assigned Team Manager | link to Team | filtered to teams for which the custom field "Show on assigned list in these tables" contains Service Request |
| Attached Files | File | Allows multiple files of any size to be attached, used for files sent or provided by the submitter |
| CR Spawned from HD ID, CR Conv ID, CR ID, CR Status, Change Summary | link to Change Request | Default: The values from the record(s) in which the Spawning HD Case ID matches the value in the ID field in the current Service Request. |

| | | |
|---|---|---|
| CI Class, CI Name, CI Responsible Team, CI Tag, CI Type, CI Serial Number | link to Configuration Item | Visible only if the linked service field "additional fields to show" contains the value "CI asset fields", so only for services that have CI's associated with them |
| Clone | Action Button | Makes a copy of the Service Request, currently unused. |
| Closed By | Link to Contact: Full Name | Dynamic Default: person who changed status to Closed |
| Configuration Items for this User | Embedded Search Result | Visible only for certain Service Categories (those likely to deal with configuration items), shows all assets for which the submitter is defined as the owner |
| Convert to Change Request | Action button | Text button: "Save Changes and Copy to Change Request" Execute Action: "Convert to Change Request if first one" (if there is not already a value in the linked CR ID field – the same Service Request cannot be linked to more than one Change request) |
| Convert to Incident | Action button | Text hyperlink: "Save Changes and Copy to Incident" Execute Action:<br><br>`if (Incident ID!='NULL') {`<br>    `Can only convert to Incident once`<br>`} else {`<br>    `Convert to Incident`<br>`}`<br><br>Run Interactively, showing user results |
| CR Closed? | Choice: Yes/No | Flag for notification when linked CR closes. Not shown to users. Default: No |
| Date Assigned | Date/Time | Date the Assigned Team last changed |
| Date Closed | Date/Time | Visible if Status=Closed or Reopened, Default: the date the status was changed to Closed |
| Date Needed | Date/Time | Allows the requester to specify a date/time for the completion of the service request. |
| Date Updated | Date/Time | Default: Date the SR is last updated |
| Description | Text, 20000 | Used to describe the request in more detail |
| Downloadable Files | File | Generally used to provide files to the submitter in response to the request (as opposed to the Attached File field which |

| | | is generally used for incoming files from the submitter) |
|---|---|---|
| Employee Name | Short Text, 70 | Visible only if Additional Fields to Show (pulled in from the selected Service) contains Employee Name |
| Employee Title | Short Text, 70 | Visible if Additional Fields to Show (from linked Service record) contains Employee Title and Login |
| Escalate | Action Button | Sets Assigned Team to 2nd Level Support Team and Priority to Critical |
| I would like to reopen my Service Request | Choice: Yes/No | Visible if status=Closed or Reopened, default is No |
| Incident ID, Incident Status, Incident Summary | Link to Incidents | Automatically linked to Incident created by the Convert to Incident action button or available to look up and select an Incident to link to |
| Knowledgebase Topics | Multi-Choice: Hardware Troubleshooting, Password Resets, Desktop Applications, Custom Applications, Upgrade and Purchase Policies, Other Service Issues | Visible if Add to Knowledgebase is Yes. Requires at least one value to be selected then. |
| Select New Tasks | Link to single field from other table: Task Template | Visible if Additional Field to show (from linked Service) contains Select New Tasks. Filtered to only Template records in which the Template Status is Active and Related Service matches the Service Request's Type of Service. |
| Number of Assignees | Floating Point | Auto-incremented when the Assigned Person field is changed |
| Number of Completed Tasks | Calc on Related Table | Count of records in embedded Tasks Table whose status is Completed |
| Number of Reopens | Floating Point | Auto-incremented when the status changes from Closed to something else |
| Number of Tasks | Calc on Related Table | Count of records in the embedded Tasks table |
| Number of Teams Assigned | Floating Point | Auto-incremented when the Assigned Team changes |
| Preferred Login | Short Text | Visible if Additional Fields to show contains Employee Title and Login |

| Priority | Choice | Required |
|---|---|---|
| Publish to Knowledgebase | Choice: Yes/No | Visible if Status=Closed or Reopened , if set to Yes, will be visible in the FAQ interface for end users |
| Resolution | Text, 20000 | Holds the solution information about the service request |
| Service Category | Choice | The user first chooses this field, which filters the available linked services to those that include that category |
| Solved within SLA | Choice: Yes/No | Visible if Status=Closed or Reopened, Default value is Yes.  It is set to No by a rule if the total hours to close is greater than the appropriate hours to close for the criticality and the service (defined by linked service and priority) |
| Spawning Incident ID | Long integer field | System field used to create link to related incident |
| Staff Only Working Notes | Append Only Text | Field hidden from end users that can be used for running notes on work done to fulfill the SR |
| Standard Solution | Choice: Yes/No | Visible only if Status=Closed or Reopened, defines a set of requests that can be looked up and imported into the solution field using the lookup next to that field. |
| Status | Workflow | Set by the workflow editor |
| Submitter Name, Submitter Team, Submitter Phone, Submitter Email, Submitter Department, Submitter Cell Phone, Submitter Login, Submitter Manager | Link to Contacts | Strict link to user of system defaulting to the user who creates the record.  Staff users can populate with a different user if taking a call. |
| Summary | Short Text, 90 | Required, brief description of the service needed |
| Tasks | Related Table: Tasks | Visible if Additional Fields to Show contains Tasks Table |
| Time Description | short text: 255 | To describe a particular time entry and what was done |
| Time Spent | Elapsed Time | For technicians to enter time spent working on the problem.  When saved, the SR will automatically convert that |

| | | into a time entry linked to the SR. |
|---|---|---|
| Total Hours to Close | Elapsed Time | Visible if Status=Closed or Reopened<br><br>Default: difference between Date Created and Date Closed |
| Working Hours to Close | Elapsed Time | Visible if Status=Closed or Reopened<br><br>Elapsed time between Date Created and Date Closed excluding the time during which the Status was Pending Customer and the non-working hours of the team in the Assigned Team field |

.

## SAVED SEARCHES

| Saved Search Name | Search Description |
|---|---|
| Assigned to a Team | Assigned Team!=NULL |
| Changed from Closed | Status changed from: 'Closed' to: any value last user's modification |
| Knowledge? = Yes | Add to Knowledgebase?='Yes' |
| Mine | Submitter Name~='$global.my_full_name' |
| My Assigned Service Requests | Assigned Person='$global.my_full_name' or Assigned Team is one of my teams |
| Not Closed | Status!='Closed' |
| Published to KB | Publish to Knowledgebase='Yes' |
| Standard Solutions | Standard Solution='Yes' |
| Updated by Customer | Updated By~='$full_name' and Submitter Team~='customer' |
| Waiting For Customer Response | Status='Pending Customer' |

## RULES AND ACTIONS

## Rule: All new Service Request actions

Runs when created by any method, no saved search

### I: Assign Service Request

```
if (Give Service Team Priority over CI Team!='Yes' and Assigned Team=NULL and CI Responsible Team!=NULL) {
      Set assigned team to ci responsible team
} else if (Assigned Team=NULL and Service Responsible Team!=NULL) {
      Set Assigned team to service responsible team
} else {
      Set assigned team to 1st level support team
}
```

### I: Trigger Creation of tasks from template

```
if (New Employee Tasks!=NULL and Additional Fields to Show~='New Employee Tasks field') {
      Update Last SR ID field in linked templates
}
```

```
if (Status!='Closed') {
      acknowledge to customer
      if (Assigned Person!=NULL) {
            email assigned person of new assignment
      } else {
            Email assigned team of new SR
      }
}
```

### I: Send emails to customer and assignee

## Rule: All edit validations and other actions

Runs when edited, no selected search

### I: All edit validations by staff

```
if (Status changed from: any value to: 'Pending Customer' last user's modification and Additional Information not changed
last user's modification) {
      Must put comments for pending customer
}
if (Status changed from: any value to: 'Closed' last user's modification and Resolution=NULL) {
      must have resolution
}
```

### I: Trigger new task creation

```
if (New Employee Tasks changed from: any value to: any value last user's modification) {
      Update Last SR ID field in linked templates
}
```

## Rule: All Customer Update Actions

### I: User Update Actions

```
if (Status='Closed' and I would like to reopen my Service Request='Yes') {
      Change Closed to Reopened
      Update # of time reopened field
} else if (Status='Pending Customer') {
      Change to Updated by Customer
}
if (Assigned Person!=NULL) {
      Email assignee customer update
} else {
      Email Assigned Team of customer update
}
```

## Rule: Tasks Just Completed

Runs on edit, Search: R: Completed Tasks just changed to All

### I: Notify Assigned team or person of completed tasks

```
if (Assigned Person!=NULL) {
      Email assigned person tasks are completed
} else {
      Email Assigned Team that tasks are completed
}
```

## Rule: Status Change Actions

Runs on edit, Search: R: Status changed last user mod

### I: All Status Change Actions

```
if (Status changed from: any value to: 'Closed' last user's modification) {
      Reset reopen flag
      if (Priority='Critical') {
            if (Total Hours to Close>'$sla_critical_priority_hours_to_complete') {
                  Set "Solved within SLA" to No
            }
      } else {
            if (Total Hours to Close>'$sla_hours_to_complete') {
                  Set "Solved within SLA" to No
            }
      }
}
```

## Rule: Assigned team or person changed

Runs on edit, Search: Assigned person or team changed

### I: Notify team or person of new assignment

```
if (Assigned Person!=NULL) {
     email assigned person of new assignment
} else {
     Email assigned team of reassignment
}
if (Assigned Team changed from: any value to: any value last user's modification) {
     Increment number of teams assigned
}
if (Assigned Person changed from: any value to: any value last user's modification) {
     Increment number of assignees
}
```

## FORMATTING FOR ACTION BUTTONS

### Add Time

**Purpose:** Triggers conversion mapping Time Spent and Time Description fields onto a new Time Entry record linked to this Task then blanks those fields.

**On Click:** Execute action.

**Action:**

#### I: Validate and Create Time Entry

```
if (Time Spent=NULL) {
     Time spent must not be blank
} else {
     Convert to Time Entry
     Blank out time fields in SR
}
```

### Clone

**Purpose:** Makes a copy of the Service Request, currently unused.

**On Click:** Execute action.

**Action:**

#### C: Clone to Service Request

**Conversion:** Service Request

### Convert to Change Request

**Purpose:** Convert to create a new Change Request, then save and close record.

**On Click:** Execute action then save record.

**Action:**

#### I: Convert to Change Request If First One

189

```
if (CR ID!=NULL) {
      can only convert to cr once
} else {
      Convert to Change Request
}
```

## Convert to Incident

**Purpose:** Validates that Problem has not been converted to Incident before, then saves and converts to an Incident.

**On Click:** Execute action then save record.

**Action:**

### I: Convert to Incident If First Conversion

```
if (Incident ID!=NULL) {
      Can only convert to Incident once
} else {
      Convert to Incident
}
```

## Escalate

**Purpose:** Sets Assigned Team to 2nd Level Support Team and Priority to Critical.

**On Click:** Execute action.

**Action:**

### U: Set Assigned Team to 2nd Level Support Team and Priority to Critical

**Updated Fields:** Assigned Team, Priority

**Updated Values:** 2nd Level Support Team, Critical

# STEPS TABLE

## FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|
| All Steps in this Workflow | Link to selected fields from other table:Steps Allow multiple values:Yes, fast search Allow entries not in source table:No | Pulls in all steps from same workflow, so we can total them to know the correct totals |
| Assigned Person | Link to selected fields from other table:People Allow multiple values:No | Filtered to people whose Teams contain the Assigned Team. Value list display. |

| | Allow entries not in source table:No | |
|---|---|---|
| Assigned Team | Link to selected fields from other table:Teams<br>Allow multiple values:No<br>Allow entries not in source table:No | |
| Image ID | Short Text | Set by formula to step number . Total Workflow Steps. Used to match to the right progress bar image. Needs to update whenever steps change. |
| Next Assigned Person, Next Assigned Team, Next Step ID, Next Step Number, Next Step Title | Link to selected fields from other table:Steps<br>Allow multiple values:No<br>Allow entries not in source table:No | Strict link, links to the Next step in the same workflow. |
| PB Image ID, Progress Bar Image | Link to selected fields from other table:Progress Images<br>Allow multiple values:No<br>Allow entries not in source table:No | Pulls in the progress bar image for the step it is on. Visible only if Workflow Type is Sequential. |
| Previous Assigned Person, Previous Assigned Team, Previous Step ID, Previous Step Number, Previous Step Title | Link to selected fields from other table:Steps<br>Allow multiple values:No<br>Allow entries not in source table:No | Strict link, links to the previous step in the same workflow. |
| Status | Choice | Used in rules to prevent the deletion of a step when a workflow is in use. Default: Deletable |
| Step Number | Integer | Minimum: 1, Maximum 10, Required |
| Step Title | Short Text | Required, maximum 90, display 45, plain text, summary field. |
| Total Steps in Workflow | Calculation on Multiple Linked Records : | Count on all Steps in the Workflow |
| Workflow Related To, Workflow Status, Workflow Title, Workflow Type | Link to selected fields from other table:Workflows<br>Allow multiple values:No<br>Allow entries not in source table:No | Required, strict link, always update. |

## Edit: All Edit by Web

**Purpose:** Refreshes search for "All Steps in this Workflow" on the record.

**Runs:** On edit.

**Actions:**

### U: Update All Steps Table

**Updated Field:** All Steps in this Workflow

**Updated Value:** Overwrite existing value with search: "MT: Steps: workflow title matches Contract Workflow Title", choosing all records that match.

## Create: All Create Actions – Set as Next Step in Previous Step, Validate Number

**Purpose:** Keeps step indicator fields updated on creation, automatically updates previous step with currently created step's ID, if one does not already exist.

**Runs:** On Creation by Web

**Actions:**

### U: Update All Steps Table

**Updated Field:** All Steps in this Workflow

**Updated Value:** Overwrite existing value with search: "MT: Steps: workflow title matches Contract Workflow Title", choosing all records that match.

### U: Update All Steps through the Workflow

**Updated Fields:** All Steps in this Workflow, Total Steps in Workflow(recalculate)

**Updated Values:** Overwrite existing value with search: "MT: Steps: workflow title matches Contract Workflow Title", choosing all records that match; [TRUE]

### U: Update Image ID in All Related Steps

**Updated Field:** Image ID

**Updated Value:** $step_number + ":" + format("step.total_steps_in_workflow",$total_steps_in_workflow)

### I: All Create Actions

```
if (Previous Step ID!=NULL) {
      Set Next Step ID for previous step record
}
```

## Create/Edit: All Validation Actions by Web

**Purpose:** Validates that workflow is in Status of "Available" before a step can be modified or created.

**Runs:** On creation or edit by web.

**Actions:**

**I: All Validation Actions by Web**

```
if (Workflow Status='In Use') {
        Not Allowed to Create or Modify Step
}
```

# SUPPORT CASES TABLE

## FIELDS

*Required fields are denoted by an * next to the field label*

| Field Label | Data Type/ Length/Default | Comments |
|---|---|---|
| *Customer Name<br>*Customer Email<br>Customer Phone<br>User Company<br>Customer Login<br>Customer Team<br>Supported User<br><br>Company Account Rep<br><br>Cell Phone | Link to selected fields from Contact table:<br><br>Name, Email, Direct phone, Company | *Defaults to the person who submits ticket* |
| Additional Notes | Append only text | For communicating with customer and allowing the customer to provide updates |
| Assigned Person | link to selected fields in contact table: full name | filtered to members of assigned team |
| Assigned Team, Team Leader | link to team table selected fields | |
| Attached Files | File | Allows multiple files of any size to be attached |
| Billable Time | Link to Billing | |
| Change Category<br>Change Summary<br>CR Conv ID<br>CR ID<br>CR Spawned from | Link to selected fields from: Change Requests | Fields to help with conversion and linking |

| | | |
|---|---|---|
| SC ID<br><br>CR Status | | |
| Closed by | link to selected fields in contact: full name | person who changed the status to Closed |
| Conv ID | Short Text | $id |
| Convert to Change Request | Action Button | Runs conversion if Change Request for this Support Case does not already exist. |
| Date Assigned | Date/time | Date the assigned person field changes |
| Downloadable Files | File | Allows attached files sent to the customer with the solution, that are also available through the FAQ interface |
| FAQ Feedback | Append Only Text | |
| I Would Like To Reopen My Ticket | Choice: yes, no | *Defaults to No*<br><br>This is used to flag a ticket for reopening and further action |
| Linked Company, Contract Status, Expiration Date, Incidents Remaining, Required Response Time, Support Hours, Type of Support | link to company fields: company name, contract status, expiration date, incidents remaining, required response time, support hours, type of support | Default: match against Customer Company field |
| Priority | Choice: Critical, High, Medium, Low, Very Low | *Defaults to Low* |
| Problem Description | text 10,000 90 x 6 default field | |
| Published | Choice: Yes, No | *Defaults to No* |
| Rating Posts | Integer | *Defaults to 0*<br><br>Used to count number of feedbacks on FAQs |
| Solution | Text, 10000 90 x 6 | For providing a solution before closing a ticket |
| Staff only Notes | Append only text field | For communicating internally – this field is not visible to end users |

| Standard Solution | Choice: Yes, No | |
|---|---|---|
| Status | Choice: Open, Assigned, Closed, Sent to Customer, Updated by Customer, Reopened | These choices represent states in the workflow field |
| Steps to Reproduce | Text, 100000 72 x 2 | |
| Summary | default field, short text 90 | |
| Time Created | Time | Time record is created |
| Type of Issue | Choice: Bug, Enhancement Request, Documentation, Installation Issue, Question, Unknown | |
| Weighted Average FAQ Rating | Integer | *Defaults to 0*<br><br>Used if you use a rating scale for FAQ feedback |

## RULES AND ACTIONS

### Assignee Change by someone else

**Purpose:** Informs assignee when assignee is changed

**Runs:** Edit by Email, Web, or API

**Search Filter:** Assignee change

**Priority:** 1

**Actions:**

#### E: Email assignee

**Template:** Case #$id assigned to you: "$summary"

### SC – All new case actions

**Purpose:** Inform assignee or assigned team of new case.

**Runs:** Create by Email, Web, or API

**Search Filter:** None

**Priority:** 2

**Actions:**

#### I: Email Assigned team or person of new case

```
if (Assigned Person!=NULL) {
      email assignee new case
} else if (Assigned Team!=NULL) {
      Email Assigned team new case
}
```

## SC – All Customer Update Actions

**Purpose:** Handles status changes and email notices from interactions with customer.

**Runs:** Edit by Email or Web

**Search Filter:** R: Updater Team is customer team

**Priority:** 3

**Actions:**

### I: Customer Update Ticket Actions

```
if (I Would Like To Reopen My Ticket changed from: any value to: 'Yes' last user's modification and Status='Closed') {
    Change Status To Reopened
}
if (Status='Sent to Customer') {
    Set State to Updated by Customer
}
if (Assigned Person!=NULL) {
    email assignee update
} else {
    Email Assigned Team of Customer Update
}
```

## FORMATTING FOR ACTION BUTTONS

## Convert to Change Request

**Purpose:** Check to see if Change Request already exists for this Support Case. If not, converts to create a Change Request from Support Case.

**On-Click:** Do nothing, execute actions, then save and open record for view.

**Action:**

### I: Convert to change request if not yet done

```
if (CR ID!=NULL) {
    can only convert sw once
} else {
    Convert to Change Request
}
```

## SUPPORT CASE SAVED SEARCHES

Active Default searches are detailed below.  The My Assigned search is the search used by the Inbox.

| Saved Search Name | Search Description |
|---|---|
| Closed Cases | Finds all records where the Status is "Closed" |
| Closed in past month | Finds all records where the Status field changed do "Closed" within the last month |

| Closed incident type of support | Finds all records where the Type of Support is "Number of incidents" and the Status changed to "Closed" in the last user modification |
|---|---|
| Closed Updated by Customer | Finds all closed records that were last updated not by the assignee but by the end user or customer |
| Created in past month | Finds all records where the creation date is over or equal to one month ago |
| My Assigned Support Cases | Finds all records assigned to me or my team. |
| Open Cases | Finds all records where the Status is "Open" |
| My Feedback Needed | Finds all records where the user is the Customer and Status is set to Sent to Customer |

## TASKS TABLE

### FIELDS

*Required fields are denoted by an * next to the field label*

*(No fields on the Tasks table are currently marked as required.)*

| Field Label | Data Type | Comments |
|---|---|---|
| Actual Working Hours to Complete | Elapsed Time | Set to the difference between Date Created and Date Completed according to the working hours of the assigned team. |
| Add Time | Action Button | Triggers conversion mapping Time Spent and Time Description fields onto a new Time Entry record linked to this Task, then blanks those fields. |
| Alert Color | Choice | Helper field to change color in view using automation. |
| All Time Spent | Calc on linked records: Time Entries | showing total time spent |
| Assigned Person | Link to single field from Employee table: Full Name | Filtered to person whose Teams field contains the Assigned Team value |
| Assigned Team | link to selected fields: Teams | Filtered to teams available for the Tasks table |
| Attached Files | File with Versioning | Converted from task template, allows multiple files, not ocr or versioning. |
| Contact Name | Link to selected fields from other table User | Person on whose behalf the task is |

| Cell Phone Company Name Direct Phone Email | table: Cell Phone, Company Name, Direct Phone, Email, Full Name | created |
|---|---|---|
| Created By Creator Name Creator Login Creator Team | Link to selected fields from other table Contact table: Full Name<br><br>Link to selected fields from Contact table: Full Name, Login, Primary Team | Automatically populated at record creation |
| Date Created | Date/Time | Automatically populated when a record is created |
| Date Done | Date/Time | Set by an action button to the current date/time |
| Date Updated | Date/Time | Automatically populated when the record is edited |
| Description | Text, 5000 | For details about the task |
| Estimated Hours to Complete | elapsed time | can be used to set the due date automatically (used for template tasks) |
| Hours to Complete | Floating Point | Required field, Default 48 |
| ID | Auto-Increment | Auto-incremented by the system; this is the record's ID number |
| Next Template IDs | Link to single field from other table: Task | Used in automation to chain tasks. |
| Opportunity Company Product | Link to selected fields from Opportunity table: Company, Opportunity Name, Product | Hidden field, used when using tasks for sales automation |
| Preceding Task ID, Preceding Task Summary | Link to selected fields from other table: Tasks | Used in automation to chain tasks. |
| Project Company, Project ID, Project ID Text, Project Manager Team, Project Category, Project Manager, | Link to selected fields from other table: Projects | Used to keep track of what project this task is related to. |

| Project Name, Type of Project, Project Status | | |
|---|---|---|
| Related To | choice: Service Request, Independent | default Service Request |
| Set Date Done | Action button | Sets the Date Done field to current date/time |
| SR Assigned Person<br><br>SR Assigned Team<br><br>SR CI Class<br><br>SR CI Name<br><br>SR CI Tag<br><br>SR CI Type<br><br>SR ID<br><br>SR Status<br><br>SR Submitter Name<br><br>SR Summary | Link to selected fields from: Service Request | Provides details on the Service Request this Task was assigned on, if one exists. |
| Status | Choice: Done, Waiting for Others, Incomplete | Defaults to Incomplete<br><br>This is the Task workflow state |
| Task Summary | Short Text, 90 | Summary field, used to describe task |
| Sales Task Type | Choice: Phone Call, Online Meeting, Onsite Demo, Field Call, Note, Task, Email | *Defaults to Phone Call*<br><br>For tracking task type |
| Template Assign Based On, Source Template ID, Template Number of Hours to Due Date, Source Task Preceding Template ID | Link to selected fields from other table: Task Templates | Keeps track of template Task spawned by. |
| Template Status | Choice | Keeps track of whether Task Template is Active or Inactive. |
| Time | Short Text | To enter description when entering |

| Description | | time spent on a task |
|---|---|---|
| Time Entries | Related table: Time Entries | Shows all time entries related to this task |
| Time Spent | Elapsed Time | To enter time spent on the task |
| Updated By | Link to selected fields from Contact table: Full Name | Automatically populated when the record is edited |
| Working Notes | Append only | For adding notes about the completion of the task |

# RULES AND ACTIONS

## Create: All create validations

**Purpose:** Restricts addition of tasks to legal statuses and specifies that due dates must be later than the day created.

**Runs:** Create by Web.

**Search Filter:** None

**Priority:** 1

**Actions:**

### I: All create validations

```
if (Project Status='Completed' or Project Status='Cancelled') {
      Cannot add task to completed or cancelled project
}
if (Date Due<this day) {
      warn that due date is in the past
}
```

## TB: Set alert color field

**Purpose:** Updates helper Alert Color field to red when criteria met.

**Runs:** Daily

**Search Filter:** Status not done or cancelled and due date is passed

**Priority:** 30

**Actions:**

### U: Set Alert Color

**Fields Changed:** Alert Color

**New Value:** Red

## Create: All Creation Actions

**Purpose:** Sets due dates, assigns project manager and teams, and emails assignees of assignation.

**Runs:** Create by Email, Web, API:

**Search Filter:** None

**Priority:** 4

**Actions:**

### I: All Creation Actions

```
if (Related to='Service Request') {
        if (Template Assign Based On='SR Assignee') {
                Set Assigned person and team to SR assignee
        }
        if (Template Number of Hours to Due Date>0 and Date Due=NULL and Status!='Done' and Source Template ID!=NULL) {
                Set Due Date to Date Created plus Number of Working Hours to Due Date
        } else if (Date Due=NULL) {
                Set Date Due to Now + 2 Days
        }
}
if (Related to='Project') {
        if (Template Assign Based On='Project Manager') {
                Set Assigned Team to PM Primary Team and Assigned Person to PM
        }
        if (Source Template ID!=NULL and Template Number of Hours to Due Date>0) {
                Set Due Date to Date Created plus Number of Working Hours to Due Date
                // checking that it started from a template prevents overwriting of manually entered due date entered when ad hoc task is created without a template
        }
        Refresh task calculation fields in project
}
```

### I: Email Assignees About New Task

```
if (Assigned Person!=NULL) {
        Email Assigned Person New Task
} else if (Assigned Team!=NULL) {
        Email Assigned Team New Task
}
```

## Edit: Trigger next tasks when completed

**Purpose:** Updates previous Task ID fields to begin automation for next tasks.

**Runs:** Edit by API

**Search Filter:** R: Status changed to Done or Cancelled

**Priority:** 20

**Actions:**

### I: Set Trigger in Next Templates

```
if (Next Template IDs!=NULL) {
        Push Project Fields to next templates
}
```

## Edit: Actions when closing or cancelling a task

**Purpose:** Performs cleanup of Alert Color, Assigned Person, Next Template IDs, and emails Project manager that project has closed or cancelled.

**Runs:** On Edit by Email or Web

**Search Filter:** R: Status changed to Done or Cancelled

**Priority:** 2

**Actions:**

### I: Edit actions when closing or cancelling

```
if (Status='Done' and Source Template ID!=NULL) {
      Set Next Template IDs
}
if (Assigned Person=NULL) {
      Set Assigned Person to person who closed
}
if (Alert Color='Red') {
      Set Alert Color to Default
}
if (Project Manager!=NULL and Project Manager!='$updated_by') {
      Email PM when task is closed or cancelled
}
```

## TB: (Disabled) Notify of upcoming task

**Purpose:** Notifies appropriate parties of pending tasks.

**Runs:** Daily

**Search Filter:** R: Upcoming Tasks

**Priority:** 3

**Actions:**

### I: Notify assigned person or team of due date

```
if (Assigned Person!=NULL) {
      Notify Assignee of Pending Activity
} else {
      Email Assigned Team of Pending Due Date
}
```

## FORMATTING FOR ACTION BUTTONS

## Add Time

**Purpose:** Validates that Time Spent has a value, then converts Time Spent and Time Description to a Time Entry and blanks out those fields.

**On Click:** Save record, then execute action.

**Action:**

```
if (Time Spent='NULL') {
     Time spent must not be blank
} else {
     Convert to Time Entry
     Set Time Spent and Time Description to Blank
}
```

## Set Date Done

**Purpose:** Sets the Date Done field to current date/time.

**On Click:** Execute action.

**Visibility:** Status is Done.

**Action:**

U: Set Date Done to Now

**Updated Field:** Date Done

**Updated Value:** NOW()

## TASK TEMPLATES TABLE

### FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|
| All Project IDs Converted | Text | |
| All SR IDs converted | Text | |
| Assign Task Based On | Choice | Dependent choices based on Related To. No default. |
| Assigned Person | Link to single field from other table:People<br>Allow multiple values:No<br>Allow entries not in source table:No | Visible only if Assign Based on has a value of Assigned Team in Task Template; filtered to people on the Assigned Team. |
| Assigned Team | Link to selected fields from other table:Teams<br>Allow multiple values:No<br>Allow entries not in source table:No | Filtered to teams whose "Show in toolbar" field includes Task. No default value, value list display. Visible only if Assign Based on has a value of Assigned Team in Task Template |

| Attached Files | File with Versioning | allows multiple files to be converted, no versioning or ocr |
|---|---|---|
| Description | Text | |
| Last Project ID text | Short Text | |
| Last SR ID | Link to single field from other table:Service Requests Allow multiple values:No Allow entries not in source table:No | |
| Last SR ID Text | Short Text | |
| Latest Project ID, Latest Project Manager, Latest Project Status | Link to selected fields from other table:Projects Allow multiple values:No Allow entries not in source table:No | |
| Latest Task ID | Link to selected fields from other table:Tasks Allow multiple values:No Allow entries not in source table:No | Mapped automatically from a completed task when the "next task templates" are updated, so that it can be used in the conversion that creates the next task as the "preceding task id" in the newly created tasks. |
| Number of Working Hours to Due Date | Floating Point | default: 18 for two working days of 9 hours |
| Preceding Template ID, Preceding Template Name | Link to selected fields from other table:Task Templates Allow multiple values:No Allow entries not in source table:No | |
| Project Type ID, Project Type | Link to selected fields from other table:Project Types Allow multiple values:No Allow entries not in source table:No | all values that are Active, visible only if Related to is Project |

| | | |
|---|---|---|
| Related to | Choice | |
| Required or Optional | Choice | |
| Service ID, Related Service | Link to selected fields from other table:Services Allow multiple values:No Allow entries not in source table:No | |
| Task Summary | Short Text | |
| Template Status | Choice | |
| zAll Linked Project IDs Used | Link to single field from other table:Projects Allow multiple values:Yes, fast search Allow entries not in source table:No | |
| zEstimated Hours to Complete | Elapsed Time | |

## RULES AND ACTIONS

### Edit: Generate Tasks when triggered from Service Requests

**Purpose:** Generate tasks and perform upkeep on All SRs converted field.

**Runs:** Edit by Web or API

**Search Filter:** R: Latest SR ID changed and not yet converted

**Priority:** 2

**Actions:**

#### C: Convert to Task

**Conversion:** Task

**Options:** Silently with no confirmation.

#### U: Copy SR ID text into All SRs converted

**Field Changed:** All SR IDs converted

**Value Changed to:** $last_sr_text +","

### Edit: Generate new tasks when triggered from Project

**Purpose:** Generate tasks and perform upkeep on converted projects.

**Runs:** Edit by Web or API

**Search Filter:** Project ID has changed and not yet converted

**Priority:** 1

**Actions:**

### C: Convert to Task

**Conversion:** Task

**Options:** Silently with no confirmation.

### U: Copy Project ID text into all projects converted

**Field Changed:** All Project IDs Converted

**Value Changed to:** $last_project_id_text +","

## WORKFLOWS TABLE

### FIELDS

| Field label | Data Type | Admin notes |
|---|---|---|
| Add Workflow Steps | Action button | Saves record and reopens it for edit. Steps table is not visible on creation, so it becomes visible after the record is reopened, so people don't forget to save the workflow before adding steps. |
| Approval Steps | Related Table : Steps | |
| Approvals Needed | Link to selected fields from other table:Approvals Allow multiple values:Yes, fast search Allow entries not in source table:No | Used if workflows are used to define parallel as well as sequential rules. This field can be used to define which approvals are needed for a given workflow and that will auto-generate the approval records. |
| Contract Party Types | Multi-Choice | Same Choices as for Contract field, to filter which workflows are available for which kinds of Contract. Required, used as filter within Contract for Workflows. Visible only if Used in= Contracts |
| Contract Types | Multi-Choice | Same choices as in the Contract table, used as a filter for workflows within Contracts. Required. Visible only if Used in = Contracts |
| Contracts using this Workflow | Related Table : Contracts | All Contracts linked to this workflow. |

| ID | Auto-Increment | |
|---|---|---|
| Number of Contracts in Draft or Pending Approval | Calculation on Multiple Linked Records : Contracts | This field will show the number of contracts that are in a status of Draft or Pending Approval. This will be used in a rule to change the status of a workflow from Available to In Use. |
| Number of Steps | Calculation on Multiple Linked Records : Steps | Count of number of step records |
| Refresh Times | Integer | Set to 0 by default. Each time a step is deleted, it is incremented, and that runs a rule to refresh the steps count and image id |
| Refresh Totals and Images | Action button | |
| Related To | Choice | Choices: Change Requests, Contracts, Documents, Service Request; Required |
| Saved | Choice | Used to make visible or hide the "Add Workflow Steps" button. It has a No value until the record is saved by clicking that button. |
| Starting Assigned Person, Starting Assigned Team, Starting Step ID, Starting Step Number | Link to selected fields from other table:Steps Allow multiple values:No Allow entries not in source table:No | Identifies the Starting Step Assignments so that a record can be sent back to the beginning. |
| Status | Choice | Used to prevent steps from being modified or deleted once a workflow has active contracts associated with it. Values: Available, In Use, Inactive. If a workflow is In Use, its steps cannot be modified. |
| Type | Type of the SW object | |
| Workflow Title | Short Text | Unique, required, maximum 90, 50 x 1 display, no default value. |
| Workflow Type | Choice | Choices: Sequential, Parallel, default value is Sequential. |

## RULES AND ACTIONS

### Edit: By Web or API – When Number of Contracts Changes

**Purpose:** Handles Status so that workflows that are in use cannot be deleted.

**Runs:** On Edit.

**Saved Search:** R: Sequential Number of Pending Contracts Just Changed

**Actions:**

### I: Status Update

```
if (Number of Contracts in Draft or Pending Approval>=1 and Status!='In Use') {
    Set Status to In Use
    if (Workflow Type='Sequential') {
        Set Linked Steps to UnDeleteable
    }
} else if ((Number of Contracts in Draft or Pending Approval=0 or Number of Contracts in Draft or Pending Approval=NULL)
and Status='In Use') {
    Set Status to Available
    if (Workflow Type='Sequential') {
        Set Linked Steps to Deletable
    }
}
```

## Edit: Validations – Sequential Workflow: Prevent Status Change

**Purpose:** Prevent users from changing Status while workflow is In Use.

**Runs:** Edits by web.

**Saved Search:** R: Workflow type is sequential

**Actions:**

### I: Edit Validations

```
if (Status changed from: 'In Use' to: 'Available' last user's modification) {
    Cannot change from In Use if currently used
}
```